Inferring bias and uncertainty in camera calibration

Annika Hagemann · Moritz Knorr · Holger Janssen · Christoph Stiller

Abstract Accurate camera calibration is a precondi-
tion for many computer vision applications. Calibration
errors, such as wrong model assumptions or imprecise
parameter estimation, can deteriorate a system’s overall
performance, making the reliable detection and quan-
tification of these errors critical.
In this work, we introduce an evaluation scheme to cap-
ture the fundamental error sources in camera calibra-
tion: systematic errors (biases) and uncertainty (vari-
ance). The proposed bias detection method uncovers
smallest systematic errors and thereby reveals imperfec-
tions of the calibration setup and provides the basis for
camera model selection. A novel resampling-based un-
certainty estimator enables uncertainty estimation un-
der non-ideal conditions and thereby extends the classi-
cal covariance estimator. Furthermore, we derive a sim-
ple uncertainty metric that is independent of the cam-
era model. In combination, the proposed methods can
be used to assess the accuracy of individual calibrations,
but also to benchmark new calibration algorithms, cam-
era models, or calibration setups. We evaluate the pro-
sposed methods with simulations and real cameras.

1 Introduction

Many applications in 3D computer vision rely on pre-
cisely knowing the camera’s mapping from the 3D world
to the 2D image. This mapping \( p : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \) is ob-
tained during camera calibration. Errors in the cali-
bration will lead to false assumptions on the mapping,
which can impact all subsequent inferences and deteri-
orate a system’s overall performance [22,36,4,40,6,1].
The detection and prevention of errors is therefore a
critical aspect of the calibration.

To ensure high accuracy calibration, recent advances
have focused on easily applicable calibration solutions.
Starting with the seminal work by Zhang in 2000 [39],
many contributions have been made with regard to im-
proving and simplifying the calibration process [18,11,
29,17]. Multiple open-source toolboxes provide directly
applicable functions for camera calibration, including
the detection of calibration markers, different camera
models, or calibration setups. We evaluate the pro-
sposed methods with simulations and real cameras.
would be critical to ensure high accuracy calibrations, but also to benchmark new calibration setups or solutions.

In general, the error sources of camera calibration can be divided into systematic errors (bias) and uncertainty (variance) [10, p.116][12]. As in any data modeling problem, systematic errors occur if a chosen model is not sufficiently flexible or imposes false assumptions on the data (Fig. 1B). In camera calibration, systematic errors can be caused by a camera projection model not being able to reflect the true geometric camera characteristics, e.g. if lens distortions are neglected. But also other sources, such as an uncompensated rolling shutter [21], or non-planarity of the calibration target [15] can result in systematic errors.

A high uncertainty, on the other hand, describes that model parameters could not be estimated reliably based on the available data. In general, uncertainties occur if the data is subject to noise and the amount of data is limited (Fig. 1C). In camera calibration, high uncertainties are commonly caused by a lack of images used for calibration, bad coverage in the image, or a non-diversity in calibration target poses [34][23][26][25].

In this paper, we address the challenge of quantifying both, systematic errors and uncertainty, in target-based camera calibration. Our goal is to reliably detect and quantify both types of errors and to condense the result to easily interpretable measures. Extending our recent work on evaluation metrics [12], we provide four main contributions (Fig. 1):

- A method to detect systematic errors (biases) in camera calibration. The method is based on estimating the observational noise in marker detections and thereby disentangles random from systematic errors in the calibration residual.
- A method to estimate parameter uncertainties under non-ideal conditions. We show that the standard estimator for the covariance matrix underestimates the uncertainty under non-ideal conditions and propose a resampling approach for reliable estimation.
- A method to predict the expected mapping error (EME) in image space, which quantifies the uncertainty (variance) in model parameters in a model-independent way.
- A detailed comparison of existing methods to quantify bias and uncertainty.

The main advantages of our approach towards quantifying systematic errors and uncertainty are that they are (a) applied as post-processing and thus build upon already captured data and (b) abstract from the underlying camera model, resulting in comparable results across different calibrations. We evaluate the proposed methods with both simulations and real cameras.

2 Related Work

2.1 Detecting systematic errors

Detecting systematic errors in camera calibration poses a challenge, because unlike the calibration of other measurement devices, there is typically no ground-truth device to compare with. Instead, camera characteristics are typically inferred indirectly through observations of well-known 3D objects.

The most common approach to detect systematic errors is by inspection of reprojection errors, i.e. the difference between predicted and observed image coordinates on the calibration dataset [7][20][22]. As the observational noise is typically assumed to be normally distributed, the reprojection errors (i.e. the residuals) should also follow a Gaussian distribution. To detect systematic errors, the two-dimensional distribution of residuals can be visualized and deviations from the expected Gaussian distribution are an indicator for systematic errors [2].

To put this qualitative comparison into numbers, recent work proposed the KL-divergence between a 2D normal distribution, and the empirical distribution of reprojection error vectors as a measure of biasedness [30].
More precisely, it was proposed to compute the median KL-divergence over a 50x50 grid across the image, where low values indicate a large similarity to the Gaussian (low bias) and high values indicate dissimilarities (higher bias).

In addition to the distribution of residuals, the magnitude of residuals is a common indicator for systematic errors. A common approach is to compare the root mean squared error (RMSE) or reconstruction result against expected values obtained from earlier calibrations or textbooks [16]. However, the magnitude of residuals varies for different cameras, lenses, calibration targets, and marker detectors and therefore only allows capturing large errors in general.

Professional photogrammetry often makes use of precisely manufactured and highly accurate 3D calibration bodies [24]. Images captured from predefined viewpoints are then used to perform a 3D reconstruction of the calibration body, where different length ratios and their deviation from the ground truth are compared against empirical data. While these methods are both, highly accurate and repeatable, they are often not feasible or too expensive for typical research and laboratory settings and require empirical data for the camera under test.

2.2 Quantification of uncertainty

The uncertainty of a calibration can, in general, be quantified by the (co-)variance in estimated model parameters \( \hat{\xi} \). As calibration is typically performed as a least squares estimation, the standard estimator for the covariance matrix \( \Sigma_{\hat{\xi},\text{std}} \) is given by an approximated backpropagation of the observational noise in the data,

\[
\Sigma_{\hat{\xi},\text{std}} = \hat{s}_d^2 (J_{\text{calib}}^T J_{\text{calib}})^{-1},
\]

where \( J_{\text{calib}} \) is the Jacobian of calibration residuals and \( \hat{s}_d^2 = \text{MSE}_{\text{calib}}/(1 - \frac{N}{N_p}) \) is the estimated accuracy, obtained from the mean squared error of the calibration MSE_{\text{calib}}, the number of parameters \( N_p \) and the number of observations \( N \) [15] p.92-96 [13] p.141-142.

Since the covariance matrix is high dimensional and its interpretation is non-trivial, it is typically reduced to a scalar metric. Typical choices are the trace of the covariance matrix \( \Sigma_{\hat{\xi}} \) or the maximum index of dispersion \( \Sigma_{\hat{\xi}} \). However, given the variety of camera models, from a simple pinhole model with only three parameters, up to local camera models with around \( 10^5 \) parameters [23,30], parameter variances are difficult to interpret and not comparable across camera models.

To address this issue, the parameter’s influence on the mapping can be considered. The metric maxERE [25] propagates the parameter covariance to image space by means of a Monte Carlo simulation. More precisely, a 5x5 grid of 3D points is projected into the image with a sampled set of model parameters. This yields a distribution of image coordinates for each grid point. The value of maxERE is then defined by the standard deviation of the most uncertain grid point. The observability metric [22] uses an analytical approach and weights the uncertainty in estimated parameters with the parameters’ influence on the mapping. This is achieved by linearly approximating the influence of parameter errors on a model cost function. Importantly, this model cost function takes into account that errors in intrinsic parameters can partially be compensated by a change of the camera coordinate system (i.e. by adjusting the extrinsics). The observability metric is then defined by an increase in calibration cost in the least observable parameter direction, where low observability values correspond to high uncertainty.

While both of these metrics provide valuable information about the uncertainty, there are some shortcomings in terms of how uncertainty is quantified. The observability metric does not consider the whole uncertainty, but only the most uncertain parameter direction. Furthermore, it quantifies uncertainty in terms of an increase in the calibration cost, which can be difficult to interpret. maxERE quantifies uncertainty in image space and is thus easily interpretable. However, it relies on a Monte Carlo Simulation instead of an analytical approach and it does not incorporate potential compensations of errors in the intrinsics by a change of the coordinate system.

A final metric to assess the accuracy of a calibration are the reprojection errors on a test dataset [25,31]. As in machine learning, testing errors significantly higher than training errors (calibration residuals) indicate an overfit which is directly related to an uncertainty in model parameters. While a testing error is a valuable measure for the accuracy of a calibration, it requires capturing a full additional test dataset. In practice, this overhead can rarely be afforded which is why we focus on metrics that rely only on the calibration data.

3 Calibration framework

3.1 Camera projection modeling

From a geometric perspective, cameras project points in the 3D world to a 2D image [13]. This projection can be expressed by a function \( p : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \) that maps a 3D point \( \mathbf{x} = (x, y, z)^T \) from a world coordinate system to a point \( \mathbf{u} = (\bar{u}, \bar{v})^T \) in the image coordinate system.
The projection can be decomposed into a coordinate transformation from the world coordinate system to the camera coordinate system \( x \mapsto x_c \) and the projection from the camera coordinate system to the image \( p_C : x_c \mapsto u_c \):

\[
\begin{align*}
\bar{u} &= p(x, \theta, \Pi) = p_C(x_c, \theta) = p_C(Rx + t, \theta),
\end{align*}
\]

where \( \theta \) are the intrinsic camera parameters and \( \Pi \) are the extrinsic parameters describing the rotation \( R \) and translation \( t \). For a plain pinhole model, the intrinsic parameters are the focal length \( f \) and the principal point \((c_x, c_y)\), i.e. \( \theta = (f, c_x, c_y) \). For this case, the projection \( p_C(x_c, \theta) \) is given by

\[
\bar{u} = f/z_c \cdot x_c + c_x,
\]

\[
\bar{v} = f/z_c \cdot y_c + c_y.
\]

To account for lens distortions, more complex models are needed. In the following, we will consider a standard pinhole camera model with varying numbers of radial distortion parameters \( k_1, k_2, k_3 \) and different focal lengths \( f_\ell, f_y \)

\[
\begin{align*}
\bar{u} &= f_\ell/z_c \cdot x_c + c_x, \\
\bar{v} &= f_y/z_c \cdot y_c + c_y.
\end{align*}
\]

where

\[
r = \sqrt{\left(\frac{u}{z_c}\right)^2 + \left(\frac{v}{z_c}\right)^2}.
\]

For more wide-angled lenses, we use the OpenCV fisheye model [19].

### 3.2 Calibration

Our methods build upon target-based camera calibration, in which planar targets are imaged in different poses relative to the camera (see Fig. 1C). Without loss of generality, we assume a single chessboard-style calibration target and a single camera in the following. The calibration dataset is a set of images \( \mathcal{F} = \{\text{frame}_j\}_{j=1}^{N}\). The chessboard calibration target contains a set of corners \( \mathcal{C} = \{\text{corner}_i\}_{i=1}^{N_c} \). The geometry of the target is well-defined, thus the 3D coordinates of chessboard-corner \( i \) in the world coordinate system are known as \( x_i = (x_i, y_i, z_i)^T \). The image coordinates of chessboard-corners in each image are determined by a corner detection algorithm [33], giving the observations \( u_i = (u_i, v_i)^T \). Depending on the corner detector, the perspective, the blur in the image, and many other factors, the observed coordinates \( u_i \) will deviate from the true image points \( \bar{u}_i \) by a certain error, called observational noise. This error is typically assumed to be independent identically distributed (i.i.d.) \( \epsilon_d \sim \mathcal{N}(0, \sigma_d^2 I) \) with a certain detector variance \( \sigma_d^2 \). Previous work has used a more elaborate description of the detector error by taking into account the angle under which the corner is observed [23], but we will stick to the most commonly used i.i.d. description.

Following [39], parameters are estimated by minimizing a calibration cost function, defined by the sum of squares of reprojection errors

\[
e_d^2 = \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{C}} ||u_{ij} - p(x_{ij}, \theta, \Pi_j)||^2.
\]

For the sake of simplicity, we present formulas for non-robust optimization here. To reduce the impact of potential outliers, we advise robustification e.g. by using a Cauchy kernel. Optimization is performed by a non-linear least-squares algorithm, which yields parameter estimates \((\hat{\theta}, \hat{\Pi}) = \arg\min(e_d^2)\). To evaluate the calibration result, one of the most common metrics is the root mean squared error (RMSE) over all \( N \) individual corners coordinates (observations) in the calibration dataset \( \mathcal{F} \) [33, p.133]:

\[
\text{RMSE}_{\text{calib}} = \sqrt{\frac{1}{N} \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{C}} ||u_{ij} - p(x_{ij}, \hat{\theta}, \hat{\Pi}_j)||^2},
\]

or its squared version, the mean squared error \( \text{MSE}_{\text{calib}} = \text{RMSE}^2_{\text{calib}} \). Note that the number of observations \( N \) is twice the number of visible corners, i.e. \( N = 2N_c N_{\mathcal{F}} \) if the board was fully visible in all images.

The covariance matrix in estimated model parameters \( \hat{\xi} = (\hat{\theta}, \hat{\Pi}) \) is theoretically given by the backpropagation of the covariance of the corner detector:

\[
\Sigma_{\xi \xi} = (J^T_{\text{calib}} \Sigma_{\epsilon d \epsilon d} J_{\text{calib}})^{-1} = \sigma_d^2 (J^T_{\text{calib}} J_{\text{calib}})^{-1},
\]

where \( \Sigma_{\epsilon d \epsilon d} = \sigma_d^2 I \) is the covariance matrix of the corner detector and \( J_{\text{calib}} \) is the Jacobian of calibration residuals [13, p.141-142]. As the variance of the corner detector \( \sigma_d^2 \) is not known a priori, it is typically replaced by the accuracy estimate \( \hat{s}_d^2 = \text{MSE}_{\text{calib}}/(1 - \frac{N_c P}{N_{\mathcal{F}}}) \) [10, p.92-96], giving

\[
\hat{\Sigma}_{\xi \xi, \text{std}} = \hat{s}_d^2 (J^T_{\text{calib}} J_{\text{calib}})^{-1}.
\]

The covariance matrix contains the (co-)variances of both, the intrinsic parameters \( \hat{\theta} \) and extrinsic parameters \( \hat{\Pi}_j \) for each image \( j \). The covariance matrix of intrinsic parameters \( \Sigma_{\hat{\theta} \hat{\theta}} \) can then be extracted as a submatrix of the full covariance matrix.

### 4 Detecting Systematic Errors: The Bias Ratio

The major challenge in quantifying systematic errors is the fact that the residuals are always a superposition of observational noise and potential systematic errors. As
the magnitude of observational noise depends on multiple factors, including the camera, blur in the image, the calibration setup and many more, it is not known a priori. Consequently, the residuals do not provide direct information on the amount of bias.

In the following, we derive a novel method to disentangle these two contributions and thereby quantify the fraction of systematic error in the mean squared reprojection error $\text{MSE}_{\text{calib}}$ of a calibration. Following the assumptions made in Section 3.2, $\text{MSE}_{\text{calib}}$ can be formulated as a superposition of random errors, caused by the observational noise (detector variance $\sigma_d^2$), and a systematic error contribution. Taking into account the number of parameters and the number of observations, one finds asymptotically (by augmentation of [13, p. 136])

$$\text{MSE}_{\text{calib}} = \sigma^2_d (1 - \frac{N_P}{N}) + \epsilon_{\text{bias}}^2 (1 - \frac{N_P}{N}) \quad \text{asymptotically}$$

where $N_P$ is the total number of free intrinsic and extrinsic parameters and $\epsilon_{\text{bias}}$ denotes the bias introduced through systematic error. The detector variance $\sigma^2_d$ is generally camera-dependent and not known a priori. Thus, to disentangle random and systematic error contributions to $\text{MSE}_{\text{calib}}$, the detector variance $\sigma^2_d$ must be determined independently.

The rationale behind many calibration approaches, and in particular guided calibration, is to find most informative camera-target configurations. To estimate $\sigma^2_d$, we propose the opposite. We explicitly use configurations which are less informative for calibration but at the same time also less likely to be impacted by systematic errors. Such an uninformative configuration is given if a calibration target only covers small local image regions, as a large fraction systematic errors can then be compensated by adjusting the target pose.

To obtain such an uninformative configuration without capturing additional data, we decompose the calibration target virtually into several smaller calibration regions, as a large fraction systematic errors can then be compensated by adjusting the target pose. Exemplary results of the BR for a simulated C(6) camera, when calibrating with models of increasing complexity. The BR clearly indicates the systematic error when using insufficiently complex models (C(3), C(5)).

#### Notes

1. Note, that the normalization of $\epsilon_{\text{bias}}$ is different to [12], to clarify its relation to the estimated accuracy $\hat{\sigma}^2_d$. The definition of the bias ratio, however, remains unaffected.

2. Here, we assume the underlying distribution is Gaussian but might be subject to sporadic outliers. The MAD multiplied by a factor of 1.4826 gives a robust estimate for the standard deviation [23].
Finally, as a simple metric between zero and one, we compute the bias ratio as

$$BR = \frac{\hat{\epsilon}^2_{\text{bias}} (1 - \frac{N_v}{N})}{\text{MSE}_{\text{calib}}}.$$  \hspace{1cm} (12)

The bias ratio is close to zero for unbiased calibration and close to one if the results are dominated by systematic errors. In combination, the bias ratio BR and the amount of bias $\hat{\epsilon}_{\text{bias}}$ provide an intuitive measure for the systematic error in a calibration.

Note, that this method is closely related to an F-test for regression model comparison [8]. By decomposing the calibration target, we are constructing a much more complex model. If there was no bias in the calibration, i.e. if the data was already explained without decomposing the target, the re-optimizations would only result in a further compensation of the observational noise. The reduction in the mean squared error (MSE) would then be fully explained by the larger number of free parameters. On the other hand, if the calibration was biased, the reduction in MSE would be larger than expected by merely adding unnecessary parameters. This additional reduction in MSE is the basis for computing the amount of bias $\hat{\epsilon}_{\text{bias}}$ and the bias ratio.

Generally, this kind of analysis can be performed for any separable calibration target. In practice, one may also choose to cite $\sqrt{BR}$, as linear quantities may be easier to interpret than the quadratic BR. The algorithm is summarized below.

**Bias Ratio: Practical implementation**

| Step | Description |
|------|-------------|
| 1.   | Perform robust camera calibration and extract a robust estimate of MSE$_{\text{calib}}$ and the optimal parameters $\hat{\theta}$ and $\hat{\Pi}$. |
| 2.   | Compute the residuals for all $v \in V$:  
- Decompose the calibration targets in each image into $N_v$ exclusive virtual targets.  
- Optimize their pose independently leaving $\hat{\theta}$ unchanged. |
| 3.   | Compute a robust estimate of the MSE over all residuals and determine $\hat{\sigma}_d^2$ using (10). |
| 4.   | Use $\hat{\sigma}_d^2$ to estimate the bias contribution $\hat{\epsilon}_{\text{bias}}$ via (11). |
| 5.   | Finally, compute the bias ratio as $BR = \hat{\epsilon}_{\text{bias}}^2 (1 - \frac{N_v}{N})/\text{MSE}_{\text{calib}}$. |

$^3$ The decomposition of the target must lead to an overdetermined estimation problem.

### 5 Uncertainty Estimation

The second type of error, in addition to biases, are errors caused by a high uncertainty in estimated model parameters. The two major challenges in quantifying uncertainty are (i) the reliable estimation of the uncertainty and (ii) the formulation of parameter uncertainties as a model-independent and easily interpretable metric. In the following, we will address both challenges. First, we propose a resampling method to reliably estimate the covariance matrix of model parameters even under non-ideal conditions. Subsequently, we derive the model-independent uncertainty metric EME that predicts the expected error in image space.

#### 5.1 Resampling-based uncertainty estimation

In general, the uncertainty of a calibration can be quantified by the covariance matrix of estimated model parameters. As parameters are estimated via nonlinear least squares, the standard estimator for the covariance matrix is a backpropagation of the detector variance (Eq. 5). While this estimator is unbiased in theory and in simulations, we observed that for real data, it tends to underestimate the uncertainty of a calibration. Comparing (a) the variance in estimated model parameters across multiple calibrations with (b) the average predicted variance, the latter tends to be smaller (see Figs. 3a, 4). This effect is especially detectable in the presence of small systematic errors, including imperfections of the calibration target or unmodeled lens distortion. As such small deviations from the ideal assumptions are almost inevitable in practice, a more robust estimator for the covariance matrix is needed to reliably estimate the uncertainty of a calibration. In the following, we will derive a resampling method to estimate uncertainty and we will propose an approximation of the method that is less computationally costly.

#### 5.2 Bootstrapping Method

We propose to apply bootstrapping [7], a nonparametric statistical technique, to obtain a more robust estimate of the covariance matrix (see Fig. 3b). Instead of relying on specific assumptions on the distribution of the data and the resulting analytical expressions, bootstrapping relies only on the data itself. The main assumption is that the dataset (sample) is representative for the population. Parameter variances and confidence intervals are then obtained by repeated resampling of the data.
5.3 Approximated Bootstrapping Method

Our results indicate that the bootstrap method provides a reliable estimate of the covariance matrix (see Section 7.2). However, a major drawback lies in the computation time, as the entire calibration has to be conducted \( n_{\text{BS}} \) times.

To reduce the computation time, we propose an approximated bootstrapping method (aBS) which re-uses the already computed Jacobian of calibration residuals. Instead of conducting the entire calibration with each of the bootstrap samples, the aBS method only conducts the last iteration of the nonlinear optimization, as we will explain in detail below.

The bundle adjustment performed during camera calibration relies on non-linear least squares estimation. Optimal parameters are found iteratively by performing so-called Gauss-Newton steps:

\[
(J^T J) \Delta \theta = J^T r
\]
\[
\theta^{k+1} = \theta^k + \Delta \theta
\]

where \( J \) is the local Jacobian at \( \theta^k \) and \( r \) are the residuals at the current iteration \( k \). Note, that for calibration, an augmented system is typically used (e.g. Levenberg-Marquardt), but for the aBS method we omit the augmentation and use the plain Gauss-Newton formulation. At the end of the calibration, the optimal parameters \( \hat{\theta} \), the local Jacobian \( J \) and the residuals \( r \) are known, where the \( k \)-th row of the Jacobian corresponds to the local derivatives \( k \)-th residual.

This is where the approximated bootstrapping method is applied: As for the original bootstrap method, we construct \( n_{\text{BS}} \) bootstrap samples \( \{F_{\text{BS,I}}\}_{i=1,...,n_{\text{BS}}} \) by sampling with replacement from the original set of images. For each of these bootstrap samples \( F_{\text{BS,I}} \), the Jacobian and the residual vector are re-composed, such that they contain only the observations of \( F_{\text{BS,I}} \). Thus, for each bootstrap sample, we get a re-composed pair
Fig. 4 Validation of resampling based uncertainty estimation. For an ideal simulation, the standard method (std), the bootstrapping method (BS) and the approximated bootstrapping method (aBS) provide similar estimates for the uncertainty and the corresponding EMEs coincide with the average true error in image space (horizontal line). When simulating an underfit, the standard method significantly underestimated the uncertainty, while the estimates of both resampling methods remained close to the average error. For real cameras, the results were similar to a simulated underfit. The standard method underestimated the uncertainty, while BS and aBS remained close to the true error. For comparability, same bootstrap samples were used for the BS and aBS method.

\((J_{BS,l}, r_{BS,l})\). For instance, if the bootstrap sample \(F_{BS,l}\) contained the first image of the original dataset twice, the pair would be given by

\[
J_{BS,l} = \begin{pmatrix} J_1 & J_1 \\ J_2 \\ \vdots \\ J_{N_F} \end{pmatrix}, \quad r_{BS,l} = \begin{pmatrix} r_1 \\ r_1 \\ \vdots \\ r_{N_F} \end{pmatrix},
\]

where the entry \(J_j\) describes the rows of the Jacobian that correspond to image \(j\) and \(r_j\) is the residual vector corresponding to image \(j\).

To obtain the bootstrap estimates \(\hat{\theta}_{BS,l}\), only the last Gauss-Newton step is conducted:

\[
(J_{BS,l}^T J_{BS,l}) \Delta \theta_{BS,l} = J_{BS,l}^T r_{BS,l}
\]

\[
\hat{\theta}_{BS,l} = \hat{\theta} + \Delta \theta_{BS,l}.
\]

Having performed this last step for each bootstrap sample, we again get a distribution of bootstrap estimates \(\{\hat{\theta}_{BS,l}\}_{l=1,...,n_{BS}}\) and the covariance matrix \(\hat{\Sigma}_{\hat{\theta},aBS}\) can be estimated as in Eq. 2.

Thus, in short, the aBS simplifies the BS method by reusing the Jacobian and residual vector of the original calibration, which are typically costly to compute.

5.4 The Expected Mapping Error

Given a reliable estimate of the parameter covariance matrix, the second challenge lies in the formulation of the uncertainty in a model-independent and easily interpretable manner. We will now derive the expected mapping error (EME), a novel uncertainty metric which quantifies the expected difference between the mapping of a calibration result \(pc(x, \hat{\theta})\) and the true (unknown) model \(pc(x, \bar{\theta})\).

Inspired by previous works [6,25], we quantify the mapping difference in image space, as pixel differences are easily interpretable. We define a set of points in image space \(G = \{u_g\}_{g=1}^{N_G}\), from which the corresponding sight rays are computed via the inverse projection \(pc^{-1}(u_g, \bar{\theta})\) using one set of model parameters \(\bar{\theta}\). Then, points on the viewing rays are backprojected to the image using the other set of model parameters \(\hat{\theta}\) [2]. The mapping error is then defined as the average squared distance between original image coordinates \(u_g\) and back-projected image points \(pc(x_g, \hat{\theta})\) (see Fig. 3):

\[
K(\bar{\theta}, \hat{\theta}) = \frac{1}{2N_G} \sum_{g \in G} ||u_g - pc(pc^{-1}(u_g, \bar{\theta}), \hat{\theta})||^2,
\]

where \(2N_G\) is the total number of image coordinates.

Formulation (21) of the mapping error assumes that an error in the intrinsics fully propagates to the image. However, the overall projection of a camera generally also includes the coordinate transformation from the world coordinate system to the camera coordinate system \(x \rightarrow x_c\). As the camera pose and therefore the extrinsics are oftentimes re-estimated in practical settings, we additionally take into account that deviations in intrinsic parameters can partially be compensated by a change in extrinsic parameters [32] and allow for a virtual compensating rotation \(R\) of the viewing rays. Thus, we formulate the effective mapping error as follows:

\[
K(\bar{\theta}, \hat{\theta}) = \min_{R} \frac{1}{2N_G} \sum_{g \in G} ||u_g - pc(R pc^{-1}(u_g, \bar{\theta}), \hat{\theta})||^2.
\]
The uncertainty metric EME is defined by the expected value of the mapping error, which can be reduced to a simple mathematical expression, as we will show below:

\[ \text{EME} = E[K(\hat{\theta}, \hat{\theta})] = \text{trace}(\Sigma_{\theta\theta}^{-1/2} H \Sigma_{\theta\theta}^{-1/2}), \]  

where \( \Sigma_{\theta\theta} \) is the covariance matrix in model parameters and \( H \) is the so-called model matrix obtained from an approximation of the effective mapping error.

In the following, we will derive expression (23). Note, that the derivation is independent of the particular choice of \( K \), provided that we can approximate \( K \) with a Taylor expansion around \( \bar{\theta} \) up to second order:

\[ K(\hat{\theta}, \hat{\theta}) \approx K(\bar{\theta}, \bar{\theta}) + \text{grad}(K)\Delta \theta + \frac{1}{2} \Delta \theta^T H K \Delta \theta \]

\[ \approx \frac{1}{2N_C} \Delta \theta^T J_{\text{res}}^T J_{\text{res}} \Delta \theta \]

\[ \approx \Delta \theta^T H \Delta \theta, \]

where \( \Delta \theta = \hat{\theta} - \bar{\theta} \) is the difference between true and estimated intrinsic parameters, \( J_{\text{res}} \) are the mapping residuals, i.e.

\[ J_{\text{res}}(\hat{\theta}, \hat{\theta}) = u_e - p_C(R p_C^{-1}(u_e), \hat{\theta}), \]

and \( J_{\text{res}} = d\text{res}/d\Delta \theta \) is the Jacobian of the residuals. Furthermore, we defined the model matrix \( H := \frac{1}{2N_C} J_{\text{res}}^T J_{\text{res}} \). For a more detailed derivation of the second step in [24] see Supplementary.

Estimated model parameters \( \hat{\theta} \) obtained from a least squares optimization are a random vector, asymptotically following a multivariate Gaussian with mean \( \mu_{\theta} = \hat{\theta} \) and covariance \( \Sigma_{\theta\theta}[27], p. 8 \). Likewise, the parameter error \( \Delta \theta = \hat{\theta} - \bar{\theta} \) follows a multivariate Gaussian, with mean \( \mu_{\Delta \theta} = 0 \) and covariance \( \Sigma_{\Delta \theta \Delta \theta} = \Sigma_{\theta\theta} \). We propagate the distribution of the parameter error \( \Delta \theta \) to find the distribution of the mapping error \( K(\hat{\theta}, \hat{\theta}) \).

In short, we find that the mapping error \( K(\hat{\theta}, \hat{\theta}) \) can be expressed as a linear combination of \( \chi^2 \) random variables:

\[ K(\hat{\theta}, \hat{\theta}) = \Delta \theta^T H \Delta \theta \]

\[ = \sum_{n=1}^{N_\theta} \lambda_n Q_n, \quad \text{with} \quad Q_n \sim \chi^2(1). \]

The coefficients \( \lambda_n \) are the eigenvalues of the matrix product \( \Sigma_{\theta\theta}^{-1/2} H \Sigma_{\theta\theta}^{-1/2} \) and \( N_\theta \) is the number of eigenvalues which equals the number of parameters \( \theta \). The full derivation of relation (25) is shown in the Supplementary. Importantly, based on expression (25), we can derive the expected value of \( K(\hat{\theta}, \hat{\theta}) \):

\[ E[K(\hat{\theta}, \hat{\theta})] = \sum_{n=1}^{N_\theta} \lambda_n E[Q_n] = \sum_{n=1}^{N_\theta} \lambda_n E[Q_n] = \sum_{n=1}^{N_\theta} \lambda_n \]

\[ = \text{trace}(\Sigma_{\theta\theta}^{-1/2} H \Sigma_{\theta\theta}^{-1/2}), \]

where we used that the \( \chi^2 \)-distribution with one degree of freedom \( \chi^2(1) \) has expectation value \( E[\chi^2(1)] = 1 \). We therefore propose the expected mapping error \( \text{EME} = \text{trace}(\Sigma_{\theta\theta}^{-1/2} H \Sigma_{\theta\theta}^{-1/2}) \) as a model-independent measure for the uncertainty.

Fig. 5 Predicting the mapping error based on parameter uncertainties. a Schematic of the derived uncertainty metric EME = trace(\( \Sigma_{\theta\theta}^{-1/2} H \Sigma_{\theta\theta}^{-1/2} \)). We define the mapping error \( K \) as the difference between two camera models in image space. Approximating \( K \) with a quadratic form in \( \Delta \theta \), we can predict its expected value by propagation of uncertainties. b Validation of the EME in simulations. The EME predicts the average true mapping error, i.e., the difference between the calibration result and the true camera model. Error bars are 95% bootstrap confidence intervals.
The expected mapping error EME can be determined for any given bundle-adjustment calibration:

1. Run the calibration and extract the optimal parameters \( \hat{\theta} \) and the Jacobian \( J_{\text{calib}} \) of the calibration cost function.
2. Compute the intrinsic parameter covariance matrix \( \Sigma_{\hat{\theta} \hat{\theta}} \) using the resampling method, or, given ideal conditions, via the classical relation (8).
3. Determine the model matrix \( H \):
   - Implement the mapping error (Eq. (22)) as a function of the parameter estimate \( \hat{\theta} \) and a parameter difference \( \Delta \theta \).
   - Numerically compute the Jacobian \( J_{\text{res}} = \frac{d \text{res}}{d \Delta \theta} \) at the estimated parameters \( \hat{\theta} \) and compute \( H = \frac{1}{2N_G} J_{\text{res}}^T J_{\text{res}} \).
4. Compute \( \text{EME} = \text{trace}(\Sigma_{\hat{\theta} \hat{\theta}} H \Sigma_{\hat{\theta} \hat{\theta}}^{-1}) \).

Note, that although the EME is measured in the same units as the mean squared error MSE_{calib}, they conceptually differ. The MSE_{calib} describes the deviation of observations in the calibration dataset from the model predictions. It can be interpreted as a training error and could be reduced by simply using less observations or more model parameters in the calibration. The EME, on the other hand, can rather be interpreted as a prediction for the testing error\(^4\). It will increase when less observations or more unnecessary model parameters are used and thereby indicate the potential overfit.

6 Experimental Evaluation

6.1 Simulations

We simulated 3D world coordinates of a single planar calibration target in different poses relative to the camera (random rotations \( \varphi_x, \varphi_y, \varphi_z \in [-\frac{\pi}{4}, \frac{\pi}{4}] \), translations \( t_x \in [0.5 \text{ m}, 2.5 \text{ m}], t_y, t_z \in [-0.5 \text{ m}, 0.5 \text{ m}] \)). We then computed the resulting image coordinates using different camera models. To simulate the detector noise, we added Gaussian noise with \( \sigma_d = 0.05 \text{ px} \) to all image coordinates. For example images, see Fig. S1.

6.2 Rendered images

By simulating marker detection, we neglect the characteristics of the real corner, but assume i.i.d. Gaussian noise in the detection. To get a better understanding of potential deviations from this assumption, and their effect on our metrics, we additionally used rendered images. During rendering, we used a ray casting technique and simulated the camera’s point spread function using multiple weighted samples per pixel. For rendered images, both the camera model and the calibration target are known and do not introduce any bias. Thereby, the full calibration pipeline, including corner detection, can be evaluated. For example images, see Fig. S1.

6.3 Evaluation with real images

We tested the methods using real images from three different lenses (see Fig. 6a for example images and Fig. S1 for more details on the lenses and the dataset). For each lens, we collected a dataset with images of a single planar calibration target. We use a coded target proposed by \( ^{33} \), to be able to associate corners across images. As no ground-truth model was available for the real lenses, we used a reference calibration\(^5\) as an approximate ground-truth to verify the evaluation methods.

7 Results

7.1 Evaluating the bias ratio

Validating the BR in simulations and experiments

To test the bias metric, we ran calibrations using camera models of increasing complexity (increasing number of non-zero intrinsic parameters), including insufficiently complex models. We then computed the robust estimate of the RMSE (median absolute deviation, MAD) and the bias ratio (BR) for each calibration. For calibrations with an insufficiently complex model, the bias ratio should detect systematic errors.

Fig. 6 shows the results of calibrations of four cameras (one rendered dataset and three real cameras shown in Fig. 6a). Each camera was calibrated with five different camera models of increasing complexity. In detail,

\(^4\) More precisely, it is a prediction of the testing error, assuming a perfect test dataset where observations homogeneously cover the image and are free of noise. In a real test dataset, a random error of feature detection within the test images would have to be added.

\(^5\) As reference, we used the average of ten calibrations with 50 random images each.
Inferring bias and uncertainty in camera calibration

Fig. 6 Experimental evaluation of the bias ratio (BR). a Exemplary rendered image and images of the same scene taken with the three different lenses used for evaluation. b Robust estimate of the RMSE (MAD, blue) and bias ratio (red) of calibrations with models of increasing complexities. For the rendered camera images, the BR drops once a model with the capability to describe the simulated mapping is employed. For all real lenses, the BR remains unexpectedly high even when using the fisheye model C(8). c Taking into account that the calibration target was not perfectly planar ($\Delta z \sim 10^{-4} \text{m}$) further reduced the BR significantly. Error bars are standard deviations of ten calibrations with 50 random images each.

The individual parameter sets are (cf. Sec. 3.1)

$\theta_{C(3)} = (f, c_x, c_y)$,

$\theta_{C(5)} = (f, f_y, c_x, c_y, k_1)$,

$\theta_{C(6)} = (f, f_y, c_x, c_y, k_1, k_2)$,

$\theta_{C(7)} = (f, f_y, c_x, c_y, k_1, k_2, k_3)$,

$\theta_{C(8)} = (f, f_y, c_x, c_y, k_1, k_2, k_3, k_4)$.

For all cameras, the MAD and BR could be reduced by using more complex camera models, as the projections were not rectilinear and thus necessitate some kind of (nonlinear) distortion modeling.

For the rendered images, the bias ratio was close to one for the first camera model C(3), indicating that the residuals were dominated by systematic errors. For the second camera model C(5), it reduced to $\text{BR} \approx 0.6$, indicating that this model was better, but not yet sufficient. When using the camera model with two radial distortion parameters C(6), which is consistent with the model that was used for rendering, the BR dropped to $\text{BR} < 0.2$. Adding additional parameters (C(7), C(8)) did not lead to a further reduction. The bias ratio thus correctly detected the presence of systematic errors for models C(3) and C(5) and indicated that one of the more complex models should be chosen for the calibration.

Surprisingly, the BR of all three real lenses remained comparatively high across all tested camera models, demonstrating that some sort of systematic error remained (Fig. 6b). Further analyses revealed that the calibration board was not perfectly planar: running the calibrations again after precisely measuring the board geometry [33] lead to a significant further reduction in systematic error (Fig. 6c). The non-planarity of the board was on the order of $10^{-4} \text{m}$ which demonstrates that even smallest imperfections in the calibration setup can be detected by the BR.

Using only the RMSE of the calibrations, some of the systematic errors could have been overlooked. For the rendered images, for instance, the RMSE already dropped to $\text{RMSE} < 0.1 \text{pixels}$ when using camera model C(5), which may have been considered sufficiently low. Likewise, even without taking into account the board-nonplanarity, the RMSE of all three real lenses was below 0.1 pixels and the calibrations may have been considered sufficiently accurate. This highlights the advantage of using the bias ratio to assess a calibration result.
For the calibrations shown in Fig. 6, the BR got close to zero, but did not reach a value of zero for any of the rendered and real cameras. This could be explained by the fact that the observational noise of the true corner detector was not perfectly i.i.d., zero-mean Gaussian distributed [9]. As the images used here were of high quality and generally showed a low observational noise ($\sigma_d \sim 0.03$ px), such small deviations from the expected Gaussian can become visible in the BR. For the real lenses, it is furthermore conceivable that some small systematic error remained, despite already taking into account the deformation of the calibration target.

Comparison of different bias metrics

We compare the bias ratio with other state-of-the-art bias metrics introduced in Section 2. We focus on the RMSE, the visualization of residuals in a smoothed 2D histogram [2] and the Kullback-Leibler divergence between a 2D normal distribution and the empirical distribution of residual vectors [30]. While [30] uses 50x50 grid across the image and computes the KL-divergence in each cell, we use a 4x4 grid, as we are working with chessboard targets and significantly less observations. We calibrated simulated cameras with high and low observational noise, using the correct camera model (C(6), unbiased), using a model with only one radial distortion parameter (C(5), weakly biased) and using a plain pinhole model (C(3), strongly biased) (see Fig. 7).

The RMSE increased with increasing amount of bias. However, as the observational noise it not known a priori, it is hard to tell whether a value of RMSE $\approx 0.4$ px reflects a correspondingly high observational noise $\sigma_d \approx 0.4$ px or whether the observational noise is lower, and the RMSE reflects a bias (Fig. 7, upper middle cell).

The 2D histogram of residuals was already more informative: visible deviations from an expected 2D Gaussian indicated a bias (Fig. 7, right column, Fig. S2). However, as the simulations show, even in the presence of biases, such deviations are not always visible (Fig. 7 middle column).

Instead of relying on visual inspection, the Kullback-Leibler divergence measures deviations from the expected Gaussian distribution: Higher values indicate higher deviations and therefore systematic errors. The KLD clearly indicated the increasing bias in the presence of high observational noise, the BR was lower, and the RMSE reflects a bias (Fig. 7, lower middle cell). In this high noise / weak bias case, the KLD remained close to zero despite the bias (see also Fig. S2 for calibrations with real lenses).

A similar effect was visible for the bias ratio: in the presence of high observational noise, the BR was lower, as it describes the fraction of systematic error. To get a full picture on the amount of systematic error, we therefore recommend to take into account both, the BR and the absolute amount of bias $\epsilon_{bias}$. The absolute amount of bias $\epsilon_{bias}$ increased consistent with the increasing amount of bias, regardless of the amount of observational noise. In combination, the BR and $\epsilon_{bias}$ were able to capture the systematic errors in all scenarios (see also Fig. S2 for calibrations with real lenses).

![Comparison of different bias metrics](image)

**Fig. 7 Comparison of different bias metrics.** Each cell shows the distribution of residuals, the RMSE, the median Kullback-Leibler divergence (KLD), the bias ratio (BR) and the absolute bias ($\epsilon_{bias}$) of the respective calibration. The results are based on simulated data, with varying degrees of observational noise and bias influences. Note the different limits of x- and y-axes. All metrics are of statistical nature, so that the precise values can vary for different calibrations. The same plots for real cameras are shown in the Supplementary.
Note, that all metrics presented here are of statistical nature, and the precise values will fluctuate depending on the dataset and the optimization, even if the same model is chosen. Note also, that the KLD metric and the 2D histogram were originally designed for calibrations with highly complex camera models and significantly more observations \[\ldots\]. It is therefore conceivable that these metrics are more powerful in such settings.

7.2 Evaluating the resampling-based uncertainty estimation

To test the resampling-based uncertainty estimation, we compared the estimated (co-)variances to the average deviation of the calibration result from the ground-truth parameters. If variances were estimated correctly, the variance of a parameter should, on average, reflect the parameter’s mean squared deviation from the true parameter.

We compared the covariance estimates of the standard method (std), the novel bootstrapping method (BS) and the approximated bootstrapping method (aBS) (Fig. 4). Using both, simulated and real images, we ran calibrations with 50 random datasets containing \(N_F = 25\) images each. For better comparability, we visualize the EME, i.e. the propagation of the covariance matrices to image space.

For an ideal simulation with no systematic errors and i.i.d. Gaussian distributed observational noise, all three methods provided similar estimates (Fig. 4, left). For all methods, the average EME was close to the true average mapping error \(\varepsilon\).

When simulating an underfit, i.e. when calibrating a camera with only one radial distortion parameter although the ground-truth camera contained two non-zero radial distortion parameters, the methods differed. The standard method significantly underestimated the uncertainty and thereby the EME. This means that the standard estimator lead to overly optimistic assumptions on the uncertainty. The BS method and the aBS method, on the other hand, remained close to the true average error.

Importantly, for the two real cameras, the results were similar to the underfit: the standard method underestimated the variance, leading to overly optimistic assumptions on the uncertainty. The BS method and the aBS method, on the other hand, remained close to the true error.\(^6\) Our results thus indicate that in practice, where small systematic errors are almost inescapable, the resampling-based methods provide more reliable estimates for the uncertainty than the standard method.

7.3 Evaluating the uncertainty metric

Validating the prediction of the EME

To validate the uncertainty metric EME in simulations, we simulated datasets with different numbers of images \((N_F \in \{10, 15, \ldots, 50\})\) and ran calibrations using a sufficiently complex camera model. Fig. 5 shows the resampling-based methods provide more reliable estimates for the uncertainty than the standard method.

Using both, simulated and real images, we ran calibrations using a sufficiently complex camera model. Fig. 5 shows that for most applications an overly pessimistic uncertainty estimates. Note, however, that for most applications an overly pessimistic error estimate is to be preferred over an overly optimistic estimate of the standard covariance estimator.

Fig. 5a,b show the same plots for two real cameras. Here, we used a reference calibration as ground-truth and ran calibrations with different numbers of images \((N_F \in \{10, 15, \ldots, 50\})\). In Fig. 5a, we used the standard covariance estimator in the computation of the EME. The EME is highly correlated with the average error, but its absolute values are significantly lower. The EME must therefore be interpreted as an upper bound for the precision that can be achieved based on a given dataset.

In Fig. 5, we used the aBS method in the computation of the EME. Using the aBS method, the EME was consistent with the average real mapping error across most calibrations. Only for calibrations with few images \((N_F \leq 15)\), the EME tended to overestimate the true error. This is caused by the fact that the theory of bootstrapping is based on asymptotic assumptions (infinite sample sizes). Although it typically works well for smaller sample sizes as well, predictions become less reliable. In the case of camera calibration, this results in overly pessimistic uncertainty estimates. Note, however, that for most applications an overly pessimistic error estimate is to be preferred over an overly optimistic estimate of the standard covariance estimator.

In summary, when using the standard covariance estimator, the EME provides an upper bound to precision that can be achieved based on a given dataset. When using the aBS method to estimate the covariance, the EME can be interpreted as an actual prediction of the mapping error given a sufficient number of calibration images.

\(^6\) Mapping error compared to the simulated ground-truth camera model.
\(^7\) Mapping error compared to the reference calibration.
Comparison of different uncertainty metrics

We compare the EME with the other state-of-the-art uncertainty metrics introduced in Section 2. We focus on \( \text{trace}(\Sigma_{\hat{\theta}}) \), maxERE \(^\text{25}\) and observability \(^\text{32}\), as these are the metrics closest to ours. Fig. 8 shows the result of all metrics for simulated calibrations with different numbers of images. To demonstrate how the uncertainty depends on both, the amount of observational noise \( \sigma_d \) and the informativeness of calibration images, we simulated three different settings (low noise / informative images, low noise / less informative images, high noise / informative images).

Fig. 8 shows the true mapping error compared to the ground truth camera model \( \sqrt{K} \), as well as all uncertainty metrics. The uncertainty, and therefore the average true error, decreased with an increasing number of images (an increasing number of observations \( N \)). As expected from theory, the scaling is approximately given by \( \Sigma_{\hat{\theta}} \propto N^{-1} \) (see also Fig. 5). Furthermore, the uncertainty was lowest for calibrations with low noise and informative images, and significantly higher for high noise or uninformative images.

All metrics were correlated with the true error. However, the metrics quantify uncertainty in very different ways: \( \text{trace}(\Sigma_{\hat{\theta}}) \) quantifies the uncertainty in model parameters, and thus inherently differs depending on the camera model. Furthermore, as its value represents the sum of different parameters variances, it does not have a unit. By re-parametrizing the camera model, the value of \( \text{trace}(\Sigma_{\hat{\theta}}) \) can change over orders of magnitude, making the comparison across different calibrations difficult.

The observability metric shows a qualitatively different scaling than the other metrics as it increases linearly with the number of images. High observabilities imply that parameters were well observable based on the data. As the EME, the observability accounts for the parameter’s effect on the mapping and for compensations of errors in the intrinsics via different extrinsics. However, it does not incorporate the full uncertainty, but only the least observable direction. Furthermore, it does not account for the observational noise, but rather evaluates whether the setup was informative. It therefore serves to evaluate a calibration setup rather than the uncertainty of the specific calibration result.

Both, maxERE and the EME quantify the expected error in image space and are thus easily interpretable. While maxERE predicts a maximum error, the EME reflects the average error. In contrast to maxERE, the EME does not require a Monte Carlo simulation.

The calibrations shown in Fig. 4 already take into account the board non-planarity.
thermore, the EME can account for a compensation via different extrinsics, which we consider a reasonable assumption in most practical scenarios.

8 Discussion and Conclusion

In this work, we derived an evaluation scheme for camera calibration, including the detection of systematic errors and the quantification of uncertainty.

We have shown that it is possible to reliably capture systematic errors by disentangling the systematic errors from the random errors in the calibration residuals. The proposed method can thereby uncover wrong model assumptions and smallest imperfections in the calibration setup. Compared to other state-of-the-art methods to detect biases, the proposed method was more reliable and more easily interpretable, as it does not involve visual inspections of error distributions, nor does it require experience or comparative values from past calibrations.

To quantify uncertainty, we proposed a resampling-based estimation of the covariance matrix. In contrast to the standard covariance estimator, the resampling method does not impose strict assumptions on the error distribution and can therefore be applied under practical, non-ideal conditions. While the standard estimator led to an underestimation of the uncertainty in non-ideal settings, the proposed method provided reliable estimates even in the presence of systematic errors. One disadvantage of the resampling method is a required minimum size of the calibration dataset, otherwise the estimation becomes overly pessimistic. This is because the theory behind bootstrapping is based on asymptotic assumptions (infinite sample sizes), which are not met with too few calibration images. In most applications, however, such overly pessimistic error estimates are to be preferred over the overly optimistic estimates of the standard method.

To reduce the computation time of the resampling method, we additionally derived an approximation of the method that completely avoids recomputation of Jacobians and residuals and is therefore significantly faster. Thereby, the resampling method becomes feasible in practice.

Finally, we have derived a model-independent and easily interpretable uncertainty metric called expected mapping error (EME). The EME quantifies the expected difference between the calibration result and the true camera in image space. By propagating the parameter uncertainty to image space, the metric is independent of the underlying camera model and comparable across different calibrations. The EME allows to quantify how informative a certain dataset is for the calibration and can thereby help to improve calibration setups.

Importantly, our derivation of the EME can also be used to predict other, application specific errors. As a generic choice, we have used the average error across the image, but one could also weight image regions differently, or define an application-specific error, such as a triangulation error of a stereo system. Furthermore, the EME can be applied for calibration guidance, in order to guide users to collect calibration datasets that explicitly minimize the uncertainty [12].

In summary, our results suggest that target-based camera calibration can be evaluated reliably by (i) computing the bias ratio to detect systematic errors, (ii) using the resampling method to estimate the uncertainty, and (iii) computing the EME to quantify the uncertainty in a model-independent manner. Together, the proposed methods can be used to assess the accuracy of individual calibrations, but also to benchmark new calibration algorithms, camera models, or calibration setups.

Acknowledgements We thank Paul-Sebastian Lauer (Robert Bosch GmbH) for supporting the experimental setup and the data acquisition.

References

1. Abraham, S., Förstner, W.: Calibration errors in structure from motion. In: Levi, P., Schanz, M., Ahlers, R.J., May, F. (eds.) Mustererkennung 1998. pp. 117–124. Springer Berlin Heidelberg, Berlin, Heidelberg (1998)
2. Beck, J., Stiller, C.: Generalized B-spline Camera Model. In: 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 2137–2142. IEEE (2018), https://ieeexplore.ieee.org/abstract/document/8500466/
3. Bodenham, D.A., Adams, N.M.: A comparison of efficient approximations for a weighted sum of chi-squared random variables. Statistics and Computing 26(4), 917–928 (2016)
4. Cheong, L.F., Peh, C.H.: Depth distortion under calibration uncertainty. Computer Vision and Image Understanding 93(3), 221–244 (Mar 2004). https://doi.org/10.1016/j.cviu.2003.09.003, https://linkinghub.elsevier.com/retrieve/pii/S1077314203001437
5. Cheong, L.F., Xiang, X.: Behaviour of SFM algorithms with erroneous calibration. Computer Vision and Image Understanding 115(1), 16–30 (Jan 2011). https://doi.org/10.1016/j.cviu.2010.08.004, https://linkinghub.elsevier.com/retrieve/pii/S1077314210001852
6. Cramariuc, A., Petrov, A., Suri, R., Mittal, M., Siegwart, R., Cadena, C.: Learning camera miscalibration detection. arXiv preprint arXiv:2005.11711 (2020)
7. Davison, A.C., Hinkle, D.V.: Bootstrap methods and their application. No. 1, Cambridge university press (1997)
Supplementary Material

**simulation**
image size: (4000 px, 4000 px)
intrinsics: $f_x = 4000$ px
    $f_y = 4100$ px
    $c_x = 2000$ px
    $c_y = 2000$ px
    $k_1 = -0.1$
    $k_2 = 0.09$

**rendered**
image size: (720 px, 720 px)
intrinsics: $f_x = 900$ px
    $f_y = 900$ px
    $c_x = 361$ px
    $c_y = 361$ px
    $k_1 = -0.4$
    $k_2 = -0.15$

**xenoplan**
image size: (1936 px, 1216 px)
intrinsics (estimated): $f_x = 3000.85$ px
    $f_y = 2163.84$ px
    $c_x = 944.92$ px
    $c_y = 597.63$ px
    $k_1 = -0.104$
    $k_2 = 0.172$
    $k_3 = -0.037$
    $k_4 = 0.046$

**cinegon**
image size: (1936 px, 1216 px)
intrinsics (estimated): $f_x = 3001.25$ px
    $f_y = 2163.96$ px
    $c_x = 970.26$ px
    $c_y = 614.11$ px
    $k_1 = -0.230$
    $k_2 = 0.533$

**manta**
image size: (1936 px, 1216 px)
intrinsics (estimated): $f_x = 851.92$ px
    $f_y = 851.95$ px
    $c_x = 947.60$ px
    $c_y = 601.86$ px
    $k_1 = 0.058$
    $k_2 = 0.058$
    $k_3 = 0.037$
    $k_4 = 0.046$

Fig. S1 Example images of datasets used for evaluation. For the real images (xenoplan, cinegon, manta), we used the same camera (Manta G-235 [38]), but different lenses (XENOPHAN 1.4/17, CINEGON 1.4/12, CINEGON 1.8/4.8).
Fig. S2 Comparison of different bias metrics. Each cell shows the distribution of residuals, the RMSE, the median Kullback-Leibler divergence (KLD), the bias ratio (BR) and the absolute bias ($\epsilon_{bias}$) of a calibration.
1 More detailed derivation of the expected mapping error (EME)

In the following, we will show a more detailed derivation of the distribution of a quadratic mapping error \( K(\hat{\theta}, \bar{\theta}) \), which quantifies the difference between the mapping of a calibration result \( p_C(x; \tilde{\theta}) \) and the true (unknown) model \( p_C(x; \bar{\theta}) \). The derivation is independent of the particular choice of \( K(\hat{\theta}, \bar{\theta}) \), provided that it can be approximated with a Taylor expansion around \( \hat{\theta} = \bar{\theta} \) up to second order:

\[
K(\hat{\theta}, \bar{\theta}) = \frac{1}{2N_g} \text{res}^T(\hat{\theta}, \bar{\theta}) \text{res}(\hat{\theta}, \bar{\theta})
\]

\[
\approx K(\hat{\theta}, \bar{\theta}) + \text{grad}(K) \Delta \theta + \frac{1}{2} \Delta \theta^T H_K \Delta \theta.
\]  

(27)

where, \( \text{res}(\hat{\theta}, \bar{\theta}) \) are the mapping residuals and \( 2N_g \) is the number of residuals. The gradient and the Hessian of \( K \) w.r.t. the parameter error \( \Delta \theta = \hat{\theta} - \bar{\theta} \) are given by

\[
\text{grad}(K) = \frac{dK}{d\Delta \theta} = 2 \frac{1}{2N_g} \text{res}(\hat{\theta}, \bar{\theta})^T J_{\text{res}}
\]

\[
H_K = \frac{d^2K}{d\Delta \theta^2} = 2 \frac{1}{2N_g} J_{\text{res}}^T J_{\text{res}} + \text{higher order terms},
\]  

(28)

where \( J_{\text{res}} = d\text{res}/d\Delta \theta \) is the Jacobian of the residuals. It follows that

\[
K(\hat{\theta}, \bar{\theta}) \approx K(\hat{\theta}, \bar{\theta}) + \frac{2}{2N_g} \text{res}(\hat{\theta}, \bar{\theta})^T J_{\text{res}} \Delta \theta + \frac{1}{4N_g} \Delta \theta^T (2J_{\text{res}}^T J_{\text{res}}) \Delta \theta
\]

\[
\approx \frac{1}{2N_g} \Delta \theta^T (J_{\text{res}}^T J_{\text{res}}) \Delta \theta
\]

\[
\approx \Delta \theta^T H \Delta \theta,
\]  

(29)

where we defined the model matrix \( H := \frac{1}{2N_g} J_{\text{res}}^T J_{\text{res}} \). Here we made the Gauss-Newton approximation \( d^2\text{res}/d\Delta \theta^2 \approx 0 \), so that the higher-order terms of the Hessian vanish [37] p. 16]. This approximation also implies that both \( J_{\text{res}} \) and \( H \) are independent of where they are evaluated [37] p.16]. In the second step we used that \( \text{res}(\hat{\theta}, \bar{\theta}) = 0 \) and thus \( K(\hat{\theta}, \bar{\theta}) = 0 \). This holds if we assume that the camera model is adequately chosen (especially not underparameterized).

We now show that for an ideal, bias-free calibration, the effective mapping error \( K(\hat{\theta}, \bar{\theta}) \) can be predicted by propagating parameter uncertainties. Estimated model parameters \( \hat{\theta} \) obtained from a least squares optimization are a random vector, asymptotically following a multivariate Gaussian with mean \( \mu_\theta = \bar{\theta} \) and covariance \( \Sigma_{\theta\theta} \). Likewise, the parameter error \( \Delta \theta \) follows a multivariate Gaussian, with mean \( \mu_\Delta \theta = 0 \) and covariance \( \Sigma_{\Delta \theta \Delta \theta} = \Sigma_{\theta \theta} \) (the quadratic cost function corresponds to the log-likelihood [37] p.9]). Finally, as \( \Delta \theta \) is a random variable, so is \( K(\hat{\theta}, \bar{\theta}) \) a random variable.

We propagate the distribution of the parameter error \( \Delta \theta \) to find the distribution of the mapping error \( K(\hat{\theta}, \bar{\theta}) \): Based on \( \Delta \theta \), we can define a multivariate standard Gaussian distributed random variable

\[
z := \Sigma_{\theta \theta}^{-1/2}(\Delta \theta - \mu_\Delta \theta)
\]

\[
\Leftrightarrow \Delta \theta = \Sigma_{\theta \theta}^{1/2} z + \mu_\Delta \theta
\]

\[
\Leftrightarrow \Delta \theta = \Sigma_{\theta \theta}^{1/2}(z + \Sigma_{\theta \theta}^{-1/2} \mu_\Delta \theta).
\]  

(30)

Plugging this into the second order approximation of \( K \) (29), we get

\[
K = \Delta \theta^T H \Delta \theta
\]

\[
= (z + \Sigma_{\theta \theta}^{-1/2} \mu_\Delta \theta) \Sigma_{\theta \theta}^{1/2} H \Sigma_{\theta \theta}^{1/2} (z + \Sigma_{\theta \theta}^{-1/2} \mu_\Delta \theta).
\]  

(31)

We now diagonalize the matrix \( \Sigma_{\theta \theta}^{1/2} H \Sigma_{\theta \theta}^{1/2} = P^T A P \) where \( A \) is a diagonal matrix containing the eigenvalues and \( P \) is an orthogonal matrix \( (P^T P = I) \) where the columns are the corresponding eigenvectors (spectral
theorem).

Plugging this into the definition of $K$, we get

$$
K = (z + \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})^T \Sigma_{\hat{\theta}^2}^{-1/2} H \Sigma_{\hat{\theta}^2}^{-1/2} (z + \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})
= (z + \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})^T P^T \Lambda P (z + \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})
= (Pz + P \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})^T \Lambda (Pz + P \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta})
= (u + b)^T \Lambda (u + b).
$$

(32)

In the last step, we replaced $b = P \Sigma_{\hat{\theta}^2}^{-1/2} \mu_{\Delta \theta}$ and $u = Pz$. As $z$ is multivariate standard Gaussian distributed and $P$ is orthogonal, $u$ will also be multivariate standard Gaussian distributed with identity covariance and expectation zero.

Finally, since $\Lambda$ is a diagonal matrix containing the eigenvalues $\lambda_n = 1, \ldots, N_\theta$ of the matrix $\Sigma_{\hat{\theta}^2}^{-1/2} H \Sigma_{\hat{\theta}^2}^{-1/2}$, this reduces to

$$
K = \Delta \theta^T H \Delta \theta
= \sum_{i=1}^{N_\theta} \lambda_n (u_n + b_n)^2 \quad \text{with} \quad u_n \sim \mathcal{N}(0,1),
$$

(33)

where $N_\theta$ is the number of eigenvalues $\lambda_n$, which equals the number of parameters (the length of the parameter vector $\theta$).

Expression (33) is a linear combination of $\chi^2$-distributed random variables. The eigenvalues $\lambda_n$ are the coefficients and the $b_n$ determine the non-centrality.

As $\mu_{\Delta \theta} = 0$, it follows that $b = 0$. $K$ is therefore just a linear combination of central first-order $\chi^2$-distributed random variables:

$$
K = \sum_{n=1}^{N_\theta} \lambda_n Q_n, \quad \text{with} \quad Q_n \sim \chi^2(1).
$$

(34)

Fig. S3 shows examples of the distribution of the mapping error $K$, predicted by a single calibration result via equation (33), as well as the observed distribution, obtained from simulating $n_r = 100$ noise realizations. The predicted and the observed distributions coincide well, which is consistent with our derivation.

There is no closed-form solution of the cumulative distribution function of the weighted sum of i.i.d. $\chi^2$-distributed random variables [3]. However, it can be approximated, e.g. with the Imhof method [14]. Furthermore, the expected value is directly accessible: The $\chi^2$-distribution with one degree of freedom $\chi^2(1)$ has expectation value $E[\chi^2(1)] = 1$. We can thus determine the expected value of the mapping error $K$:

$$
E[K(\hat{\theta}, \bar{\theta})] = \mathbb{E} \left[ \sum_{n=1}^{N_\theta} \lambda_n Q_n \right] = \sum_{n=1}^{N_\theta} \lambda_n \mathbb{E}[Q_n] = \sum_{n=1}^{N_\theta} \lambda_n
= \text{trace}(\Sigma_{\hat{\theta}^2}^{-1/2} H \Sigma_{\hat{\theta}^2}^{-1/2})
= \text{trace}(\Sigma_{\hat{\theta}^2} H).
$$

This is the proposed uncertainty metric EME (expected mapping error).
Inferring bias and uncertainty in camera calibration

Fig. S3 Predicted and observed distribution of the mapping error $K$ for simulated calibrations with $\sigma_d = 1$ px. Vertical lines show true and predicted expected value $E[K]$. With an increasing number of frames, the distribution becomes increasingly narrow (note the different x-axes). The prediction coincides well with the observed distribution.

Fig. S4 Re-scaled visualization of Fig. 8c. The uncertainty scales with the number of observations $N$, and therefore with the number of frames $N_F$ approximately as $\Sigma_{\hat{\theta}} \propto N^{-1}$. Re-scaling the metrics with the number of frames therefore gives approximately constant values. Values are means across 50 random samples, error bars are 95% bootstrap confidence intervals.
2 Application in calibration guidance

To demonstrate the practical use of the EME, we apply it in calibration guidance. Calibration guidance refers to systems that predict most informative next observations to reduce the remaining uncertainty and then guide users towards these measurements. We choose an existing framework, called calibration wizard [23] and extend it with our metric.

Calibration wizard predicts the next best pose by minimizing the trace of the intrinsic parameter’s covariance matrix \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \). This is equivalent to minimizing the sum of the parameter’s variances. However, depending on the camera model, parameters will affect the image in very different ways. High variance in a given parameter will not necessarily result in a proportionally high uncertainty in the image.

To avoid such an imbalance, we suggest to minimize the uncertainty in image space, instead of parameter space, i.e. to replace \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \) with the EME given by \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}} \Sigma_{\theta\theta}^{-1/2} \Sigma_{\theta\theta}^{-1/2}) \).

To compare the methods, we used a dataset of calibration images (see Fig. S5a). Starting with two random images, the system successively selected the most informative next image with (i) the original metric \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \), (ii) our metric \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}} \Sigma_{\theta\theta}^{-1/2} \Sigma_{\theta\theta}^{-1/2}) \) and (iii) randomly. Using the pinhole model with radial distortion, the poses suggested by \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \) and \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}} \Sigma_{\theta\theta}^{-1/2} \Sigma_{\theta\theta}^{-1/2}) \) were similarly well suited, both leading to a significantly faster reduction in uncertainty than random images (Fig. S5). However, when changing the camera model, e.g. by parameterizing the focal length in millimeters instead of pixels, simulated here by a division by 100 \( f \to 0.01 \cdot f \), the methods differed: the poses proposed by \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}} \Sigma_{\theta\theta}^{-1/2} \Sigma_{\theta\theta}^{-1/2}) \) reduced uncertainty significantly faster than \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \). This can be explained by the fact that when minimizing \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \), the variance of less significant parameters will be reduced just as much as the variance of parameters with large effect on the mapping. This example shows that the performance of \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}}) \) can be affected by the choice of the model, while \( \text{trace}(\Sigma_{\hat{\theta}\hat{\theta}} \Sigma_{\theta\theta}^{-1/2} \Sigma_{\theta\theta}^{-1/2}) \) remains unaffected.