Bias Detection and Prediction
of Mapping Errors in Camera Calibration

Annika Hagemann[1,2(✉)], Moritz Knorr[1], Holger Janssen[1],
and Christoph Stiller[2]

[1] Robert Bosch GmbH, Corporate Research, Computer Vision Research Lab,
Hildesheim, Germany
annika.hagemann@de.bosch.com
[2] Institute of Measurement and Control, Karlsruhe Institute of Technology,
Karlsruhe, Germany

Abstract. Camera calibration is a prerequisite for many computer vision applications. While a good calibration can turn a camera into a measurement device, it can also deteriorate a system's performance if not done correctly. In the recent past, there have been great efforts to simplify the calibration process. Yet, inspection and evaluation of calibration results typically still requires expert knowledge.

In this work, we introduce two novel methods to capture the fundamental error sources in camera calibration: systematic errors (biases) and remaining uncertainty (variance). Importantly, the proposed methods do not require capturing additional images and are independent of the camera model. We evaluate the methods on simulated and real data and demonstrate how a state-of-the-art system for guided calibration can be improved. In combination, the methods allow novice users to perform camera calibration and verify both the accuracy and precision.

1 Introduction

In 2000 Zhang published a paper [19] which allowed novice users to perform monocular camera calibration using only readily available components. Several works, including systems for guided calibration, improved upon the original idea [10,12,13]. However, we believe that a central building block is still missing: a generic way to evaluate the quality of a calibration result. More precisely, a way to reliably quantify the remaining biases and uncertainties of a given calibration. This is of critical importance, as errors and uncertainties in calibration parameters propagate to applications such as visual SLAM [9], ego-motion estimation [3,17,20] and SfM [1,4]. Despite this importance, typical calibration procedures rely on relatively simple metrics to evaluate the calibration, such
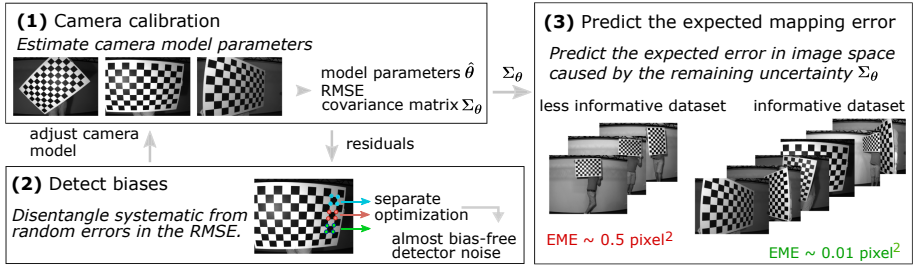
Fig. 1. Proposed camera calibration procedure, including the detection of systematic errors (biases) and the prediction of the expected mapping error.

as the root mean squared error (RMSE) on the calibration dataset. Furthermore, many frequently used metrics lack comparability across camera models and interpretability for non-expert users.

In general, the error sources of camera calibration can be divided into underfit (bias) and overfit (high variance). An underfit can be caused by a camera projection model not being able to reflect the true geometric camera characteristics, an uncompensated rolling shutter, or non-planarity of the calibration target. An overfit, on the other hand, describes that the model parameters cannot be estimated reliably, i.e. a high variance remains. A common cause is a lack of images used for calibration, bad coverage in the image, or a non-diversity in calibration target poses. In this paper, we address the challenge of quantifying both types of errors in target-based camera calibration, and provide three main contributions:

– A method to detect systematic errors (underfit) in a calibration. The method is based on estimating the variance of the corner detector and thereby disentangles random from systematic errors in the calibration residual (Fig. 1).
– A method to predict the expected mapping error (EME) in image space, which quantifies the remaining uncertainty (variance) in model parameters in a model-independent way. It provides an upper bound for the precision that can be achieved with a given dataset (Fig. 1).
– The application of our uncertainty metric EME in calibration guidance, which guides users to poses that lead to a maximum reduction in uncertainty. Extending a recently published framework [10], we show that our metric leads to further improvement of suggested poses.

In combination, these methods allow novice users to perform camera calibration and verify both the accuracy and precision of the result. Importantly, the work presented here explicitly abstracts from the underlying camera model and is therefore applicable in a wide range of scenarios. We evaluate the proposed methods with both simulations and real cameras.

2   Fundamentals

Camera Projection Modeling. From a purely geometric point of view, cameras project points in the 3D world to a 2D image [6]. This projection can be expressed by a function $\boldsymbol{p} : \mathbb{R}^3 \to \mathbb{R}^2$ that maps a 3D point $\boldsymbol{x} = (x, y, z)^T$ from a world coordinate system to a point $\bar{\boldsymbol{u}} = (\bar{u}, \bar{v})^T$ in the image coordinate system. The projection can be decomposed into a coordinate transformation from the world coordinate system to the camera coordinate system $\boldsymbol{x} \to \boldsymbol{x_c}$ and the projection from the camera coordinate system to the image $\boldsymbol{p_C} : \boldsymbol{x_c} \to \bar{\boldsymbol{u}}$:

$$\bar{\boldsymbol{u}} = \boldsymbol{p}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{\Pi}) = \boldsymbol{p_C}(\boldsymbol{x_c}, \boldsymbol{\theta}) = \boldsymbol{p_C}(\boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}, \boldsymbol{\theta}), \tag{1}$$

where $\boldsymbol{\theta}$ are the *intrinsic* camera parameters and $\boldsymbol{\Pi}$ and are the *extrinsic* parameters describing the rotation $\boldsymbol{R}$ and translation $\boldsymbol{t}$. For a plain pinhole camera, the intrinsic parameters are the focal length f and the principal point (ppx, ppy), i.e. $\boldsymbol{\theta} = (f, \text{ppx}, \text{ppy})$. For this case, the projection $\boldsymbol{p_C}(\boldsymbol{x_c}, \boldsymbol{\theta})$ is given by $u = f/z_c \cdot x_c + \text{ppx}$, $v = f/z_c \cdot y_c + \text{ppy}$. In the following, we will consider more complex camera models, specifically, a standard pinhole camera model (S) with radial distortion $\boldsymbol{\theta}_S = (f_x, f_y, \text{ppx}, \text{ppy}, r_1, r_2)$, and the OpenCV fisheye model (F) $\boldsymbol{\theta}_F = (f_x, f_y, \text{ppx}, \text{ppy}, r_1, r_2, r_3, r_4)$ [8].

Calibration Framework. We base our methods on target-based camera calibration, in which planar targets are imaged in different poses relative to the camera. 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 $\mathcal{F} = \{\text{frame}_i\}_{i=1}^{N_{\mathcal{F}}}$. The chessboard calibration target contains a set of corners $\mathcal{C} = \{\text{corner}_i\}_{i=1}^{N_{\mathcal{C}}}$. The geometry of the target is well-defined, thus the 3D coordinates of chessboard-corner i in the world coordinate system are known as $\boldsymbol{x}_i = (x_i, y_i, z_i)^T$. The image coordinates $\boldsymbol{u}_i = (u_i, v_i)^T$ of chessboard-corners are determined by a corner-detector with noise σ_d. Thus, the observed coordinates $\boldsymbol{u}_i$ are assumed to deviate from the true image points $\bar{\boldsymbol{u}}_i$ by an independent identically distributed (i.i.d.) error $\epsilon_d \sim \mathcal{N}(0, \sigma_d)$. Estimation is performed by minimizing a calibration cost function, typically defined by the quadratic sum over reprojection errors

$$\epsilon_{\text{res}}^2 = \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{C}} ||\boldsymbol{u}_{ij} - \boldsymbol{p}(\boldsymbol{x}_{ij}, \boldsymbol{\theta}, \boldsymbol{\Pi}_j)||^2. \tag{2}$$

For the sake of simplicity, we present formulas for non-robust optimization here. Generally, we advise robustification, e.g. using a Cauchy kernel. Optimization is performed by a non-linear least-squares algorithm, which yields parameter estimates $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\Pi}}) = \text{argmin}(\epsilon_{\text{res}}^2)$.

A common metric to evaluate the calibration is the root mean squared error (RMSE) over all N individual corners coordinates (observations) in the calibration dataset $\mathcal{F}$ [6, p. 133]:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{j \in \mathcal{F}} \sum_{i \in \mathcal{C}} ||\boldsymbol{u}_{ij} - \boldsymbol{p}(\boldsymbol{x}_{ij}, \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\Pi}}_j)||^2}, \tag{3}$$

The remaining uncertainty in estimated model parameters $\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\Pi}}$ is given by the parameter's covariance matrix $\boldsymbol{\Sigma}$. The covariance matrix can be computed by backpropagation of the variance of the corner detector σ_d^2:

$$\boldsymbol{\Sigma} = (\boldsymbol{J}_{\text{calib}}^T \boldsymbol{\Sigma}_d^{-1} \boldsymbol{J}_{\text{calib}})^{-1} = \sigma_d^2 (\boldsymbol{J}_{\text{calib}}^T \boldsymbol{J}_{\text{calib}})^{-1}, \qquad (4)$$

where $\boldsymbol{\Sigma}_d = \sigma_d^2 \boldsymbol{I}$ is the covariance matrix of the corner detector and $\boldsymbol{J}_{\text{calib}}$ is the Jacobian of calibration residuals [6]. The covariance matrix of intrinsic parameters $\boldsymbol{\Sigma}_\theta$ can be extracted as a submatrix of the full covariance matrix.

3   Related Work

Approaches to evaluating camera calibration can be divided into detecting systematic errors and quantifying the remaining uncertainty in estimated model parameters. Typical choices of uncertainty metrics are the trace of the covariance matrix [10], or the maximum index of dispersion [13]. 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 [2,15], 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* [12] quantifies uncertainty by propagating the parameter covariance into pixel space by means of a Monte Carlo simulation. The value of maxERE is then defined by the variance of the most uncertain image point of a grid of projected 3D points. The *observability* metric [16] weights the uncertainty in estimated parameters (here defined by the calibration cost function's Hessian) with the parameters' influence on a model cost function. Importantly, this model cost function takes into account a potential compensation of differences in the intrinsics by adjusting the extrinsics. The observability metric is then defined by the minimum eigenvalue of the weighted Hessian.

While both of these metrics provide valuable information about the remaining 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 pixel 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 differences in the intrinsics by adjusting the extrinsics.

The second type evaluation metrics aims at finding *systematic* errors. As camera characteristics have to be inferred indirectly through observations, there is a high risk of introducing systematic errors in the calibration process by choosing an inadequate projection model, neglecting rolling-shutter effects, or using an out-of-spec calibration target, to give a few examples. If left undetected, these errors will inevitably introduce biases into the application.

Historically, one way to detect systematic errors is to compare the resulting RMSE or reconstruction result against expected values obtained from earlier

calibrations or textbooks [7]. However, these values vary for different cameras, lenses, calibration targets, and marker detectors and, hence, only allow capturing gross errors in general. Professional photogrammetry often makes use of highly accurate and precisely manufactured 3D calibration bodies [11]. Images captured from predefined viewpoints are then used to perform a 3D reconstruction of the calibration body. Different length ratios and their deviation from the ground truth are then computed to assess the quality of the calibration by comparing against empirical data. While these methods represent the gold standard due to the accuracy of the calibration body and repeatability, they are often not feasible or too expensive for typical research and laboratory settings and require empirical data for the camera under test. The methods presented in the following relax these requirements but can also be seen as a complement to this standard.

4   Detecting Systematic Errors

In the following, we derive the bias ratio (BR), a novel metric for quantifying the fraction of systematic error contribution to the mean squared reprojection error (MSE). Following the assumptions made in Sect. 2 one finds that asymptotically (by augmentation of [6, p. 136])

$$\text{MSE}_{\text{calib}} = \underbrace{\sigma_d^2(1 - \frac{N_{\mathcal{P}}}{N})}_{\text{random error}} + \underbrace{\epsilon_{\text{bias}}^2}_{\text{systematic error contribution}}, \tag{5}$$

where $N_{\mathcal{P}}$ is the total number of free intrinsic and extrinsic parameters and ϵ_{bias} denotes the bias introduced through systematic errors. The variance σ_d^2 is generally camera dependent and not known a priori. To disentangle stochastic and systematic error contributions to the MSE, we need a way to determine σ_d^2 independently: The rationale behind many calibration approaches, and in particular guided calibration, is to find most informative camera-target configurations (cf. Fig. 1). For bias estimation, 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. More specifically, we decompose the calibration target virtually into several smaller calibration targets $\mathcal{V} = \{\text{target}_i\}_{i=1}^{N_{\mathcal{V}}}$, usually consisting of exclusive sets of the four corners of a checker board tile (cf. Fig. 2a). The poses of each virtual calibration target in each image are then estimated individually while keeping the camera intrinsic parameters fixed. Pose estimation is overdetermined with a redundancy of two (four tile corners and six pose parameters). From the resulting MSE values, MSE_v with $v \in \mathcal{V}$, we compute estimates of σ_d^2 via (5) assuming the bias is negligible within these local image regions

$$\widehat{\sigma}_{d_v}^2 = \frac{\text{MSE}_v}{1 - \frac{6}{8}} = 4\,\text{MSE}_v. \tag{6}$$
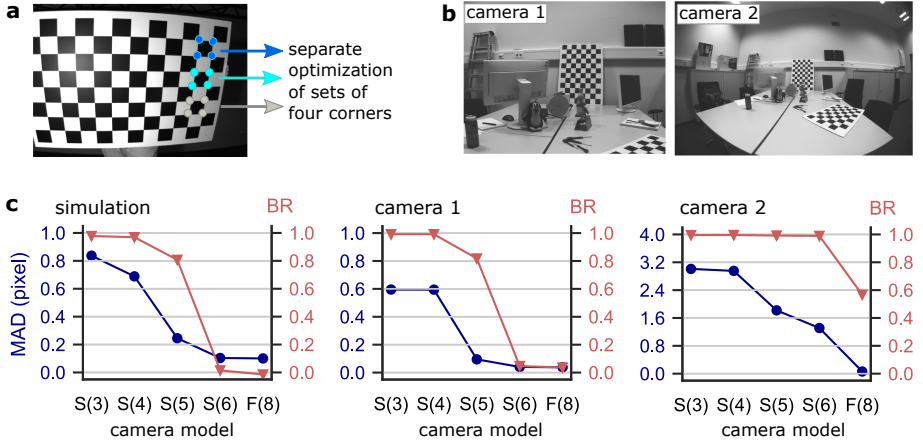
Fig. 2. Detecting systematic errors. **a** Illustration of the virtual decomposition of the calibration target into smaller targets used to estimate the corner detector variance. **b** Exemplary image of the same scene with the two test cameras. **c** Results of the bias ratio (BR) and the robust estimate of the RMSE (MAD) for one simulated and the two real cameras, using models of different complexities. For details see Sect. 7.

To obtain an overall estimate of $\widehat{\sigma}_d^2$, we compute the MSE in (6) across the residuals of all virtual targets, using the MAD as a robust estimator[1]. Finally, we use $\widehat{\sigma}_d^2$ to determine $\epsilon_{\mathrm{bias}}^2$ using (5) and compute the *bias ratio* as

$$\mathrm{BR} = \frac{\epsilon_{\mathrm{bias}}^2}{\mathrm{MSE}_{\mathrm{calib}}}. \tag{7}$$

The bias ratio is zero for unbiased calibration and close to one if the results are dominated by systematic errors. The bias ratio is an intuitive metric that quantifies the fraction of bias introduced by systematic errors. A bias ratio below a certain threshold τ_{BR} is a necessary condition for a successful calibration and a precondition for uncertainty estimation.[2] Generally, this kind of analysis can be performed for any separable[3] calibration target.

Practical Implementation. Computation of the bias ratio for target-based calibration procedures:

---

[1] 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 [14].

[2] To choose a threshold, it can be used that $\frac{1}{1-\mathrm{BR}}$ is approximately F-distributed, representing the ratio of the residual sum of squares (SSE) of the calibration over the SSE of the virtual targets, weighted by their respective degrees of freedom. However, this only holds approximately, as the datapoints are not independent. We therefore use an empirical threshold of $\tau_{BR} = 0.2$, allowing for small biases.

[3] The decomposition of the target must lead to an overdetermined estimation problem.

1. Perform robust camera calibration and extract a robust estimate of $\mathrm{MSE}_{\mathrm{calib}}$ and the optimal parameters $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\Pi}}$.
2. Compute the residuals for all $v \in \mathcal{V}$:
   – Decompose the calibration targets found in each image into a total of $N_{\mathcal{V}}$ exclusive virtual calibration targets.
   – Optimize their pose independently leaving $\hat{\boldsymbol{\theta}}$ unchanged.
3. Compute a robust estimate of the MSE over all residuals and determine $\widehat{\sigma}_d^2$ using (6).
4. Use $\widehat{\sigma}_d^2$ to determine the bias contribution $\epsilon_{\mathrm{bias}}^2$ via (5).
5. Finally, compute the bias ratio as $\mathrm{BR} = \epsilon_{\mathrm{bias}}^2 / \mathrm{MSE}_{\mathrm{calib}}$ and test the result against the threshold τ_{BR}.

5   The Expected Mapping Error (EME)

The second type of error source, in addition to biases, is a high remaining uncertainty in estimated model parameters. We will now derive a novel uncertainty metric, the *expected mapping error* (EME), which is interpretable and comparable across camera models. It quantifies the expected difference between the mapping of a calibration result $\boldsymbol{p}_C(\boldsymbol{x}; \hat{\boldsymbol{\theta}})$ and the true (unknown) model $\boldsymbol{p}_C(\boldsymbol{x}; \bar{\boldsymbol{\theta}})$.

Inspired by previous works [5,12], we quantify the mapping difference in *image space*, as pixel differences are easily interpretable: we define a set of points in image space $\mathcal{G} = \{\boldsymbol{u}_i\}_{i=1}^{N_\mathcal{G}}$, which are projected to space via the inverse projection $\boldsymbol{p}_C{}^{-1}(\boldsymbol{u}_i; \bar{\boldsymbol{\theta}})$ using one set of model parameters and then back to the image using the other set of model parameters [2]. The mapping error is then defined as the average distance between original image coordinates $\boldsymbol{u}_i$ and back-projected image points $\boldsymbol{p}_C(\boldsymbol{x}_i; \hat{\boldsymbol{\theta}})$ (see Fig. 3):

$$\tilde{K}(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) = \frac{1}{N} \sum_{i \in \mathcal{G}} ||\boldsymbol{u}_i - \boldsymbol{p}_C(\boldsymbol{p}_C{}^{-1}(\boldsymbol{u}_i; \bar{\boldsymbol{\theta}}); \hat{\boldsymbol{\theta}})||^2, \tag{8}$$

where $N = 2N_\mathcal{G}$ is the total number of image coordinates. Since small deviations in intrinsic parameters can oftentimes be compensated by a change in extrinsic parameters [16], we allow for a virtual compensating rotation $\boldsymbol{R}$ of the viewing rays. Thus, we formulate the *effective* mapping error as follows:

$$K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) = \min_{\boldsymbol{R}} \frac{1}{N} \sum_{i \in \mathcal{G}} ||\boldsymbol{u}_i - \boldsymbol{p}_C(\boldsymbol{R} \, \boldsymbol{p}_C{}^{-1}(\boldsymbol{u}_i; \bar{\boldsymbol{\theta}}); \hat{\boldsymbol{\theta}})||^2. \tag{9}$$

We now show that for an ideal, bias-free calibration, the effective mapping error $K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})$ can be *predicted* by propagating parameter uncertainties. Note that the following derivation is independent of the particular choice of K, provided that we can approximate K with a Taylor expansion around $\hat{\boldsymbol{\theta}} = \bar{\boldsymbol{\theta}}$ up to second order:

$$\begin{aligned} K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) &\approx K(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) + \mathrm{grad}(K)\boldsymbol{\Delta\theta} + \frac{1}{2}\boldsymbol{\Delta\theta}^T \boldsymbol{H}_K \boldsymbol{\Delta\theta} \\ &\approx \frac{1}{N}\boldsymbol{\Delta\theta}^T(\boldsymbol{J_{\mathrm{res}}}^T \boldsymbol{J_{\mathrm{res}}})\boldsymbol{\Delta\theta} \\ &\approx \boldsymbol{\Delta\theta}^T \boldsymbol{H} \boldsymbol{\Delta\theta}, \end{aligned} \tag{10}$$
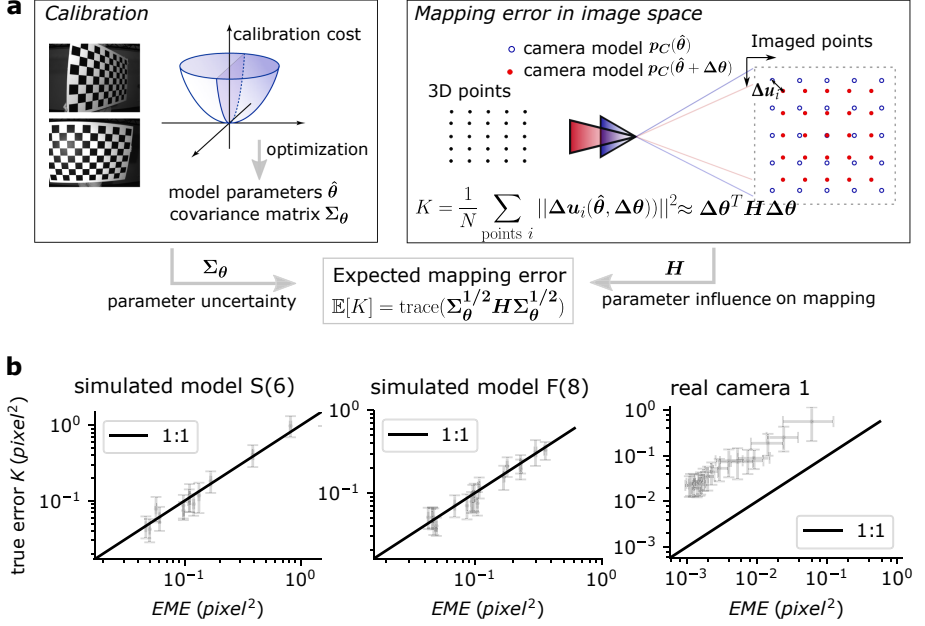
Fig. 3. Predicting the mapping error based on parameter uncertainties. **a** Schematic of the derived uncertainty metric EME = $\mathrm{trace}(\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2})$. **b** Evaluation in simulation and experiments. The simulation results validate the derived relation (5). For real cameras, the EME is a lower bound to the error, as non-ideal behavior can lead to higher absolute errors. Error bars are 95% bootstrap confidence intervals.

where $\boldsymbol{\Delta\theta} = \bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}$ is the difference between true and estimated intrinsic parameters, $\mathbf{res}_i(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) = \boldsymbol{u}_i - \boldsymbol{p_C}(\boldsymbol{R}\,\boldsymbol{p_C}^{-1}(\boldsymbol{u}_i; \bar{\boldsymbol{\theta}}); \hat{\boldsymbol{\theta}})$ are the mapping residuals and $\boldsymbol{J_{\mathrm{res}}} = d\mathbf{res}/d\boldsymbol{\Delta\theta}$ is the Jacobian of the residuals. Furthermore, we defined the *model matrix* $\boldsymbol{H} := \frac{1}{N}\boldsymbol{J_{\mathrm{res}}}^T\boldsymbol{J_{\mathrm{res}}}$. For a more detailed derivation of the second step in (10), see Supplementary.

Estimated model parameters $\hat{\boldsymbol{\theta}}$ obtained from a least squares optimization are a random vector, asymptotically following a multivariate Gaussian with mean $\boldsymbol{\mu_\theta} = \bar{\boldsymbol{\theta}}$ and covariance $\boldsymbol{\Sigma_\theta}$ [18, p. 8]. Likewise, the parameter error $\boldsymbol{\Delta\theta} = \bar{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}$ follows a multivariate Gaussian, with mean $\boldsymbol{\mu_{\Delta\theta}} = \boldsymbol{0}$ and covariance $\boldsymbol{\Sigma_{\Delta\theta}} = \boldsymbol{\Sigma_\theta}$. We propagate the distribution of the parameter error $\boldsymbol{\Delta\theta}$ to find the distribution of the mapping error $K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})$. In short, we find that the mapping error $K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})$ can be expressed as a linear combination of χ^2 random variables:

$$
\begin{aligned}
K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}) &= \boldsymbol{\Delta\theta}^T \boldsymbol{H} \boldsymbol{\Delta\theta} \\
&= \sum_{i=1}^{N_\theta} \lambda_i Q_i, \quad \text{with} \quad Q_i \sim \chi^2(1).
\end{aligned}
\tag{11}
$$

The coefficients λ_i are the eigenvalues of the matrix product $\boldsymbol{\Sigma_\theta}^{1/2} \boldsymbol{H} \boldsymbol{\Sigma_\theta}^{1/2}$ and N_θ is the number of eigenvalues which equals the number of parameters $\boldsymbol{\theta}$. The full derivation of relation (11) is shown in the Supplementary. Importantly, based on expression (11), we can derive the expected value of $K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})$:

$$\mathbb{E}[K(\hat{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})] = \mathbb{E}[\sum_{i=1}^{N_\theta} \lambda_i Q_i] = \sum_{i=1}^{N_\theta} \lambda_i \mathbb{E}[Q_i] = \sum_{i=1}^{N_\theta} \lambda_i$$
$$= \text{trace}(\boldsymbol{\Sigma_\theta}^{1/2} \boldsymbol{H} \boldsymbol{\Sigma_\theta}^{1/2}),$$

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

Practical Implementation. The expected mapping error EME can be determined for any given bundle-adjustment calibration:

1. Run the calibration and extract the RMSE, the optimal parameters $\hat{\boldsymbol{\theta}}$ and the Jacobian $\boldsymbol{J}_{\text{calib}}$ of the calibration cost function.
2. Compute the parameter covariance matrix $\boldsymbol{\Sigma} = \sigma_d^2 (\boldsymbol{J}_{\text{calib}}^T \boldsymbol{J}_{\text{calib}})^{-1}$ and extract the intrinsic part $\boldsymbol{\Sigma_\theta}$.
3. Determine the model matrix $\boldsymbol{H}$:
   – Implement the mapping error (Eq. (9)) as a function of the parameter estimate $\hat{\boldsymbol{\theta}}$ and a parameter difference $\boldsymbol{\Delta\theta}$.
   – Numerically compute the Jacobian $\boldsymbol{J}_{\text{res}} = d\mathbf{res}/d\boldsymbol{\Delta\theta}$ at the estimated parameters $\hat{\boldsymbol{\theta}}$ and compute $\boldsymbol{H} = \frac{1}{N} \boldsymbol{J}_{\text{res}}^T \boldsymbol{J}_{\text{res}}$.
4. Compute EME $= \text{trace}(\boldsymbol{\Sigma_\theta}^{1/2} \boldsymbol{H} \boldsymbol{\Sigma_\theta}^{1/2})$.

6 Experimental Evaluation

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 $t_z \in [0.5 \text{ m}, 2.5 \text{ m}]$, $t_x, t_y \in [-0.5 \text{ m}, 0.5 \text{ m}]$). We then computed the resulting image coordinates using different camera models. To simulate the detector noise, we added Gaussian noise with $\sigma_d = 0.1$ px to all image coordinates. To validate the bias ratio, we simulated a pinhole camera with two radial distortion parameters, but ran calibrations with different models, including insufficiently complex models (underfit). To validate the uncertainty measure EME $= \text{trace}(\boldsymbol{\Sigma_\theta}^{1/2} \boldsymbol{H} \boldsymbol{\Sigma_\theta}^{1/2})$, we ran calibrations with different numbers of simulated frames ($N_{\mathcal{F}} \in [3, 20]$) and $n_r = 50$ noise realizations for each set of frames. After each calibration, we computed the true mapping error K with respect to the known ground-truth (Eq. 9) and the EME.

Evaluation with Real Cameras. We tested the metrics for two different real cameras (see Fig. 2b). For each camera, we collected a total of $n = 500$ images of a planar calibration target. As reference, we performed a calibration with all 500 images. To test the bias metric, we ran calibrations with camera models of different complexities (Fig. 2c). To test the uncertainty metric EME, we ran calibrations with different numbers of randomly selected frames ($N_{\mathcal{F}} \in [3, 20]$, 50 randomly selected datasets for each $N_{\mathcal{F}}$). For each calibration, we computed both the true mapping error K with respect to the reference and the EME.

7   Results

Validating the Bias Ratio. Figure 2c shows the robust estimate of the RMSE (median absolute deviation, MAD) and the bias ratio for the calibrations of three cameras (one simulated camera and the two real cameras shown in Fig. 2b) for varying numbers of non-zero intrinsic calibration parameters, representing different camera models. In detail, the individual parameter sets are $\boldsymbol{\theta}_{S(3)} = (f, \mathrm{ppx}, \mathrm{ppy})$, $\boldsymbol{\theta}_{S(4)} = (f_x, f_y, \mathrm{ppx}, \mathrm{ppy})$, $\boldsymbol{\theta}_{S(5)} = (f_x, f_y, \mathrm{ppx}, \mathrm{ppy}, r_1)$, $\boldsymbol{\theta}_{S(6)} = (f_x, f_y, \mathrm{ppx}, \mathrm{ppy}, r_1, r_2)$, and $\boldsymbol{\theta}_{F(8)} = (f_x, f_y, \mathrm{ppx}, \mathrm{ppy}, r_1, r_2, r_3, r_4)$ (cf. Sect. 2).

   For all cameras, the MAD and BR can be reduced by using a more complex camera model which is to be expected, since the projections are not rectilinear and thus necessitate some kind of (nonlinear) distortion modeling. For the simulated camera and camera 1, a bias ratio below $\tau_{BR} = 0.2$ is reached using the standard camera model (S) with two radial distortion parameters. For camera 2, a low bias ratio cannot be reached even when using OpenCV's fisheye camera model with 8 parameters. This highlights the advantage of the bias ratio over the RMSE: the low RMSE could wrongfully be interpreted as low bias – the bias ratio of BR ≈ 0.6, however, demonstrates that some sort of systematic error remains and a more complex model should be tried.

Validating the Uncertainty Metric. To validate the uncertainty metric EME in simulations, we ran calibrations with different numbers of images using a pinhole with radial distortion S(6) and a fisheye camera F(8). Figure 3b shows the uncertainty metric EME $= \mathrm{trace}(\boldsymbol{\Sigma_\theta}^{1/2} \boldsymbol{H} \boldsymbol{\Sigma_\theta}^{1/2})$ and the real average mapping error. Consistent with Eq. (5), the EME predicts the average mapping error. For the *real* camera, the EME is highly correlated with the true mapping error, however the absolute values of the real errors are higher, which is to be expected in practice. It reflects that (i) the ground-truth is only approximated by the reference calibration, (ii) deviations from the ideal assumptions underlying the covariance matrix (Eq. (4)), and (iii) deviations from the i.i.d. Gaussian error assumption. This limitation affects all metrics that are based on the covariance matrix computed via Eq. (4). The EME therefore provides an *upper bound* to the precision that can be achieved for a given dataset.

Comparison with State-of-the-Art. We compare the EME with the other state-of-the-art uncertainty metrics introduced in Sect. 3. We focus on $\mathrm{trace}(\boldsymbol{\Sigma_\theta})$,
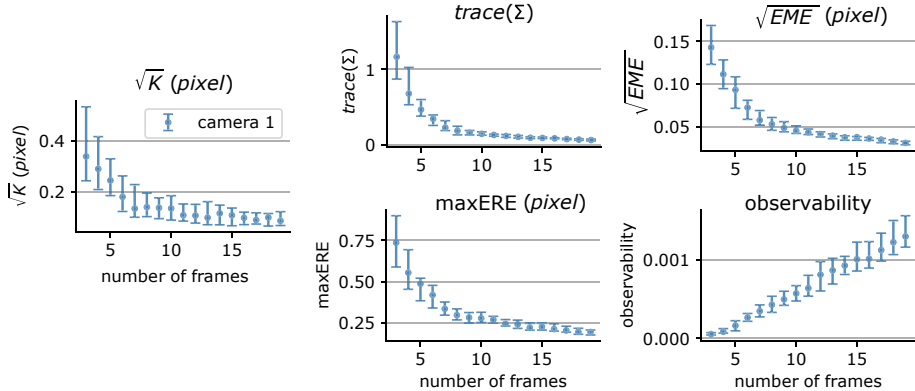
Fig. 4. Comparison of state-of-the-art metrics for real camera 1. On average, the true error K decreases with the number frames. For comparability with maxERE, we show $\sqrt{K}$ and $\sqrt{EME}$ in units of pixels. All metrics are correlated with the true error, but absolute values and the scaling differ. Values are medians across 50 random samples, error bars are 95% bootstrap confidence intervals.

maxERE [12] and *observability* [16], as these are the metrics closest to ours (Fig. 4). All metrics provide information about the remaining uncertainty and are correlated with the true error. However, the metrics quantify uncertainty in very different ways: trace($\boldsymbol{\Sigma_\theta}$) quantifies the uncertainty in model parameters, and thus inherently differs depending on the camera model. The *observability* metric accounts for the parameter's effect of the mapping and for compensations via different extrinsics. However, it does not incorporate the full uncertainty, but just the least observable direction. Furthermore, the absolute values are comparatively difficult to interpret, as they measure an increase in the calibration cost. maxERE quantifies the *maximum* expected reprojection error in image space and is therefore easily interpretable. Similar to maxERE, the EME predicts the expected error in image space and is therefore easily interpretable. Instead of a *maximum* error, the EME reflects the *average* error. In contrast to maxERE, the EME does not require a Monte Carlo simulation. Furthermore, the EME can account for a compensation via different extrinsics, which we consider a reasonable assumption in many scenarios.

8   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 exitisting framework, called calibration wizard [10] and extend it with our metric. Calibration wizard predicts the next best pose by minimizing the trace of the intrinsic parameter's covariance matrix
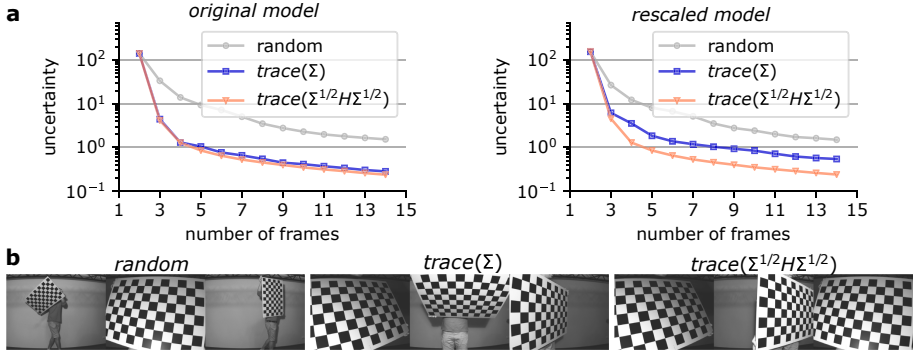
Fig. 5. Application of EME for calibration guidance. **a** For the original model, both metrics lead to a similarly fast reduction in uncertainty. Rescaling the model to a different unit of the focal length results in a reduced performance of trace($\boldsymbol{\Sigma_\theta}$), while trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$) remains unaffectd. Uncertainty is quantified by the average of the *uncertainty map* proposed by calibration wizard [10]. **b** Examples of suggested poses.

trace($\boldsymbol{\Sigma_\theta}$). 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 parameters, i.e. to replace trace($\boldsymbol{\Sigma_\theta}$) with trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$).

To compare the methods, we use images of camera 1 (see Fig. 2b). Starting with two random images, the system successively selectes the most informative next image with (i) the original metric trace($\boldsymbol{\Sigma_\theta}$), (ii) our metric trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$) and (iii) randomly. Using the pinhole model with radial distortion, the poses suggested by trace($\boldsymbol{\Sigma_\theta}$) and trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$) are similarly well suited, both leading to a significantly faster convergence than random images (Fig. 5). 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 ($f \rightarrow 0.01 \cdot f$), the methods differ: the poses proposed by trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$) reduce uncertainty significantly faster than trace($\boldsymbol{\Sigma_\theta}$). This can be explained by the fact that when minimizing trace($\boldsymbol{\Sigma_\theta}$), 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 trace($\boldsymbol{\Sigma_\theta}$) can be affected by the choice of the model, while trace($\boldsymbol{\Sigma_\theta}^{1/2}\boldsymbol{H}\boldsymbol{\Sigma_\theta}^{1/2}$) remains unaffected.

9   Conclusion and Future Research

In this paper, we proposed two metrics to evaluate systematic errors and the remaining uncertainty in camera calibration. We have shown that the *bias ratio* (BR) reliably captures underfits, which can result from an insufficiently complex

model. Furthermore, we have shown that it is possible to predict the *expected mapping error* (EME) in image space, which provides an upper bound for the precision that can be achieved with a given dataset. Both metrics are model-independent and therefore widely applicable. Finally, we have shown that the EME can be applied for calibration guidance, resulting in a faster reduction in mapping uncertainty than the existing parameter-based approach.

In future, we will extend the metrics to multi-camera systems and extrinsic calibration. Furthermore, we would like to incorporate an analysis of the coverage of the camera field of view into our evaluation scheme.

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, Heidelberg (1998). https://doi.org/10.1007/978-3-642-72282-0_11
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. Cheong, L.F., Peh, C.H.: Depth distortion under calibration uncertainty. Comput. Vis. Image Underst. **93**(3), 221–244 (2004). https://doi.org/10.1016/j.cviu.2003.09.003. https://linkinghub.elsevier.com/retrieve/pii/S1077314203001437
4. Cheong, L.F., Xiang, X.: Behaviour of SFM algorithms with erroneous calibration. Comput. Vis. Image Underst. **115**(1), 16–30 (2011). https://doi.org/10.1016/j.cviu.2010.08.004. https://linkinghub.elsevier.com/retrieve/pii/S1077314210001852
5. Cramariuc, A., Petrov, A., Suri, R., Mittal, M., Siegwart, R., Cadena, C.: Learning camera miscalibration detection. arXiv preprint arXiv:2005.11711 (2020)
6. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision (2004). https://doi.org/10.1017/CBO9780511811685. oCLC: 171123855
7. Luhmann, T., Robson, S., Kyle, S., Boehm, J.: Close-range Photogrammetry and 3D Imaging. De Gruyter textbook, De Gruyter (2013). https://books.google.de/books?id=TAuBngEACAAJ
8. OpenCV: OpenCV Fisheye Camera Model. https://docs.opencv.org/master/db/d58/group__calib3d__fisheye.html
9. Ozog, P., Eustice, R.M.: On the importance of modeling camera calibration uncertainty in visual SLAM. In: 2013 IEEE International Conference on Robotics and Automation, pp. 3777–3784. IEEE (2013). https://ieeexplore.ieee.org/abstract/document/6631108/
10. Peng, S., Sturm, P.: Calibration wizard: a guidance system for camera calibration based on modelling geometric and corner uncertainty. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1497–1505 (2019)
11. Rautenberg, U., Wiggenhagen, M.: Abnahme und ueberwachung photogrammetrischer messsysteme nach vdi 2634, blatt 1. PFG 2/2002, S.117-124 (2002). https://www.ipi.uni-hannover.de/fileadmin/ipi/publications/VDI2634_1e.pdf
12. Richardson, A., Strom, J., Olson, E.: AprilCal: assisted and repeatable camera calibration. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1814–1821. IEEE (2013). https://ieeexplore.ieee.org/abstract/document/6696595/

13. Rojtberg, P., Kuijper, A.: Efficient pose selection for interactive camera calibration. In: 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 31–36. IEEE (2018). https://ieeexplore.ieee.org/abstract/document/8613748/

14. Rousseeuw, P.J., Croux, C.: Alternatives to the median absolute deviation. J. Am. Stat. Assoc. **88**(424), 1273–1283 (1993)

15. Schoeps, T., Larsson, V., Pollefeys, M., Sattler, T.: Why Having 10,000 Parameters in Your Camera Model is Better Than Twelve. arXiv preprint arXiv:1912.02908 (2019). https://arxiv.org/abs/1912.02908

16. Strauss, T.: Kalibrierung von Multi-Kamera-Systemen. KIT Scientific Publishing (2015). https://d-nb.info/1082294497/34

17. Svoboda, T., Sturm, P.: What can be done with a badly calibrated Camera in Ego-Motion Estimation? (1996). https://hal.inria.fr/inria-00525701/

18. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment — a modern synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) IWVA 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44480-7_21

19. Zhang, Z.: A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1330–1334 (2000). https://ieeexplore.ieee.org/abstract/document/888718/

20. Zucchelli, M., Kosecka, J.: Motion bias and structure distortion induced by calibration errors, pp. 68.1–68.10. British Machine Vision Association (2001). https://doi.org/10.5244/C.15.68. http://www.bmva.org/bmvc/2001/papers/51/index.html