

Enhanced calibration of camera setups for high-performance visual odometry

Igor Cvišić, Ivan Marković and Ivan Petrović

*University of Zagreb Faculty of Electrical Engineering and Computing,
Department of Control and Computer Engineering,
Laboratory for Autonomous Systems and Mobile Robotics,
Unska 3, HR-10000, Zagreb, Croatia
{igor.cvisic, ivan.markovic, ivan.petrovic}@fer.hr*

Abstract

For robots and autonomous system that rely on visual data for operating in the real world, camera calibration is an indispensable step as it relates image information to the geometric structure of the 3D world. Although it is convenient to consider a several decades old problem as something that is swiftly solvable with a dedicated toolbox, we should still push calibration methods to their practical limits in order to gain valuable insights, and especially when robots are operating in circumstances that concern human safety. In this paper we propose a camera setup calibration procedure with emphasis on visual odometry accuracy. We focus on target-based calibration and two popular datasets are used for evaluating visual odometry and SLAM algorithms, namely the EuRoC and KITTI datasets. Our procedure consists of: (i) introducing a novel highly accurate corner detection algorithm robust to challenging illumination conditions, (ii) investigating different lens distortion models, (iii) incorporating static and dynamic board deformation models, (iv) ex-post analysis of reprojection error sensitivity and calibration parameter uncertainty, and (v) grid search method based on odometry accuracy when board poses do not constrain calibration parameters well enough. The whole process significantly reduced the reprojection error when calibrating the camera setups of the EuRoC and KITTI datasets. We tested four different odometries, namely SOFT, ORB-SLAM2, VINS-FUSION, and VISO2 – all four showed higher accuracy with the proposed calibration parameters. Moreover, with the proposed calibration method our SOFT2 scored 0.53% in translation and 0.0009 deg/m in rotation error rendering it currently the highest ranking algorithm on the KITTI scoreboard.

Keywords: Camera calibration, Visual odometry, Visual SLAM, Parametric lens models

1. Introduction

Camera calibration is an indispensable step for reliably relating image data captured by visual sensors to the real-world environment. Therefore, to be able to provide geometrical inference and correct for non-linearities in the optical system, they need to be calibrated both intrinsically and extrinsically. Given the fact that “higher-level” downstream processes, such as localization and navigation in general, rely on calibration data, the accuracy and reliability thereof is directly dependent on the results of the calibration procedure. Although it is convenient and mind assuring to consider a several decades old problem as something quickly solvable with a toolbox, we should still strive to push the methods to their practical limits when systems that depend upon them are to be used in circumstances that concern human safety. In this paper we focus on the problem of calibrating one of the most widely used visual sensors in robotics and autonomous systems, namely the stereo camera. We focus on the EuRoC and KITTI datasets, inspired by their significance in the visual odometry community and motivated by our experience of achieving the highest localization score during the EuRoC competition [1] and of currently holding the highest odometry rank on the KITTI score board [2].

Camera calibration consists of determining an optimal set of specific parameters: (i) intrinsic parameters that define the projection equation and relate 2D pixels to the 3D environment, (ii) lens distortion parameters that correct the nonlinearities introduced by the light gathering apparatus, and (iii) extrinsic parameters that define relative poses between the cameras in a multi-camera system. The underlying model is usually the pinhole camera model, while it is also worth mentioning the unified projection model that encompasses both standard cameras as well as catadioptric vision systems [3]. Lens distortions can be captured by parametric models, such as the Conrady-Brown model [4] or Kannala-Brandt model [5], or by a generic model [6] that densely relate the pixels with the 3D environment in a purely mathematical way, without offering a physical interpretation of the underlying physics. Additionally for the stereo camera setup, the extrinsic calibration also entails relating one camera’s coordinate frame to the other. The accuracy of the stereo extrinsic calibration is also of great importance as it affects the process of reasoning about the 3D world structure from a pair of simultaneously captured displaced images. Besides the fact that the calibration optimization process, when taken as a whole, involves optimizing over non-linear functions

and non-Euclidean domains, it is also a process highly sensitive to the experimental design. The design of the experiment is the one in charge of making all the parameters identifiable, i.e., observable. If the calibration procedure is not strictly defined, it is not surprising that two independent practitioners can obtain different calibration solutions for the same system (putting aside changes in the system introduced by environmental and stress factors). What can be even more puzzling, is that downstream processes will work seemingly well with different sets of parameters, until proven otherwise during the operation. Given that, some works aim at guiding the user during the experiment so that the process gathers as much as useful information as possible [7, 8].

Calibration methods can be target-based [9, 10], i.e., they can rely on a dedicated known target for calibration, and targetless [11, 12, 13], i.e., they can rely on tracking existing features in the 3D environment. The calibration with targets is carried out using a known calibration pattern, most often the checkerboard pattern, since it facilitates detection of salient features like corners, and resolves the scale ambiguity by knowing the inner corner distances. Reprojection error, i.e., the measure of distances between the detected and model reprojected corners in the image, is used as the optimization criterion. A classical calibration procedure consists of capturing multiple images of the calibration target with varying poses. Another approach is one-shot calibration which aims at calibrating a single or multiple cameras from a single image of multiple calibration targets [14, 15]. Such an approach is very practical for calibrating multiple cameras and when calibrations needs to be performed often; however, this also presents a challenge of having smaller variability in the distance and orientation of the calibrations boards. The calibration boards are typically assumed to be perfect realizations of the designed calibration pattern; however, imperfections are present in the form of printing errors, plastic deformations and dynamic deformations of the assumed planar board and researchers have investigated models that can capture such imperfections and achieve smaller calibration errors [16, 17, 18]. Calibration methods can also be offline, in the sense that calibration is performed prior to operation, or they can run online during the operation of the robot or autonomous system [19, 20]. Offline and target-based methods have the advantage of a completely controlled experimental design, while online and targetless methods are presented with the challenge of extracting useful information during limited maneuverability. Online methods are an important tool for long-term reliable operation of robots and autonomous systems, but they have not yet superseded classical target-based methods. A popular state-of-the-art framework for camera calibration is the Kalibr toolbox [21, 22, 23] offering calibration of multiple cameras, visual-inertial units, and rolling shutter cameras. Its pattern is based on AprilTags, while circle and ring planar calibration patterns can also be used [24].

An important instrument in advancing visual localization algorithms for robots and autonomous systems are public datasets as they enable evaluation and comparison of different approaches. Regarding visual odometry and SLAM, several datasets have been published over the years: the KITTI dataset

[25, 26], Málaga Urban dataset [27], KITTI-360 dataset [28], The EuRoC micro aerial vehicle dataset [29], Oxford Robotics Car dataset [30], Multivehicle Stereo Event Camera Dataset (MVSEC) [31], and a Stereo Event Camera Dataset (DSEC) [32]. Most datasets were recorded with a suite of sensors and were mounted on a vehicle driving in an urban environment, the only exception being the EuRoC dataset which is specialized for micro aerial vehicles. Regardless of the dataset, a visual camera setup requires prior calibration – ideally for each sequence when outdoor recording is performed in diverse weather conditions. Usually, datasets provide camera calibration parameters which introduces dependence on the default calibration parameters, but sometimes the datasets also includes calibration images enabling researchers to conduct calibration by themselves. The EuRoC dataset provides two separate calibration datasets recorded with a stereo setup, one with the single AprilTag target, and another with the single checkerboard target. The AprilTag dataset consist of 1449 images, where in about 1000 images whole or partial AprilTag is captured. The checkerboard dataset consists of 2376 images, where in about 2000 images whole or partial checkerboard is captured. Images are recorded at 20 Hz and neighboring images of the target are highly redundant for calibration purposes. A smaller subset of images is sufficient for calibration purposes, while the Apriltag target is preferred to the checkerboard target, since it is larger (0.66×0.66 m vs. 0.42×0.36 m), contains more corners (144 vs. 42), and has the ability to be reliably matched between stereo pairs even in the case of partial visibility due to AprilTag coded ID. Given that, AprilTag target poses yiled more constraints to the system and to more accurate parameters. The KITTI calibration dataset consists of a single shot of 12 checkerboard targets acquired with one grayscale stereo pair and one color stereo pair. Boards consist of 5×7 , 11×7 and 15×5 corners with a spacing of 10 cm. The KITTI dataset has been recorded during several days, and each day has its own corresponding dataset recorded the day after the drive. Therefore, there are 5 calibration datasets in total, each consisting of a single image. As discussed earlier, accuracy of “higer-level” downstream methods such as visual odometry and SLAM greatly depend on the accuracy of the calibration, thus having more accurate parameters can improve the overall performance of the system. For example, authors in [33] proposed to calibrate a discrete stereo deformation field above the two rectified image planes for the KITTI dataset, which would be able to correct deviations of the real camera system from the default calibration model. The authors showed that the calibrated deformation field improves the accuracy of the recovered camera motion.

In this paper we propose a camera setup calibration approach with emphasis on enhancing visual odometry accuracy. We focus on two popular datasets used for evaluating visual odometry and SLAM algorithms, namely the EuRoC and KITTI datasets. We present an improved corner detection algorithm, robust to poor lightning conditions, which detects line segments and extracts corner position with sub-pixel accuracy using line intersections. We investigate different lens models, namely the Conrady-Brown model and the Kannala-Brandt model, and also include static and dynamic board deformation modeling

in the calibration procedure. We also conduct ex-post analysis of reprojection error sensitivity and calibration parameter uncertainty. The whole process significantly reduced the reprojection error when calibrating the camera setups of the EuRoC and KITTI datasets. Finally, we tested four methodologically different odometries, namely our SOFT [1, 2], ORB-SLAM2 [34], VINS-FUSION [35, 36], and VISO2 [37] – all four showed higher accuracy with the proposed calibration parameters. Moreover, our SOFT2 with the proposed calibration method scored 0.53% in translation error and 0.0009 deg/m in rotation error rendering it currently the highest ranking algorithm on the KITTI scoreboard¹. The present paper is an extension of our preliminary work published in [38] with the following novelties: (i) improved corner detection accuracy, (ii) addition of static and dynamic board deformations, (iii) parameter uncertainty analysis, and (iv) investigation of both the EuRoC and KITTI datasets. Although our approach focuses on two specific datasets, it is still applicable to any camera setup, and we believe it to be relevant to the community due to its general insights, and relevance that EuRoC and KITTI had and probably will in the forthcoming years.

2. Proposed camera calibration setup

The primary goal of this paper is to investigate calibration approaches for obtaining a set of calibration parameters that can produce more accurate trajectories for a general visual odometry algorithm. The proposed calibration approach consists of three steps. The first step is a robust corner detection algorithm that can reliably extract corners in the calibration pattern under realistic conditions. Having the corners extracted and thus the structure of the calibration pattern, the next step is to choose a lens model that can best capture the distortion effects on the image. Furthermore, it is often the case that the target boards are imperfect due to printing errors and plastic deformations (less than ideal planar board or board leaning on the wall) and that they are carried and manipulated by hand to capture a diverse set of board poses (during this process the board can exhibit dynamic deformations that differ throughout the calibration images). To account for these effects, as the third step we exploit models proposed in [16, 18] to capture static and dynamic deformations. In the paper the experimental analysis is concentrated on the stereo cameras of the KITTI and EuRoC datasets, but the approach is applicable to multiple camera setups, as well as calibration of monocular cameras.

Given that, calibration essentially comes down to the problem of finding an optimal set of parameters p by minimizing an objective function $F(p, z)$ based on the square of residuals

$$F(p, z) = e(p, z)^T e(p, z), \quad (1)$$

where residuals are reprojection errors of detected corners locations z in the images

$$e(p, z) = z - \pi(p, X) \quad (2)$$

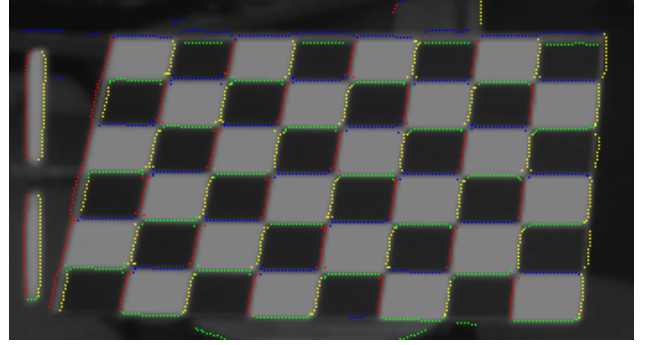


Figure 1: Detected blobs for one of the boards of the KITTI calibration sequence 2011-10-03, magnified 2×. Positive vertical edges are shown in red, negative vertical in yellow, positive horizontal in blue, and negative horizontal in green color. Best viewed in color.

with X being a 3D point and, for brevity, function $\pi(\cdot)$ captures all the previously mentioned models that participate in projecting a 3D point to the image. The set of optimal parameters, including camera intrinsics, lens distortion coefficients, and extrinsics, is then determined via

$$p^* = \underset{p}{\operatorname{argmin}} F(p, z). \quad (3)$$

The steps of the proposed calibration approach are experimentally analyzed on two popular datasets for evaluating visual odometry and SLAM algorithms in mind, namely the EuRoC and KITTI datasets.

2.1. Corner detection

Calibration images are usually recorded in indoor or outdoor environments without the possibility or much concern about the lightning which should ideally be diffuse. Most commonly, one or more direct light sources are present, and under certain angles, the board can reflect the light directly to the camera. Very often, the light reflected from the white squares is so bright that it saturates the corresponding pixels and the excess charge spills into the adjacent ones. Consequently, white squares appear larger than the black ones. Given that, instead of having a unique corner at the intersection of two edges, there can occur four edges intersecting at two different locations (as can be seen on examples in Fig. 3). We address this issue with the observation that opposite edges share the same slope, but differ in the offset. Therefore, in order to correctly capture this geometry, we propose to fit two parallel lines into the two subsets of edge points – one line for the edge of one black square, and a parallel line into the other. In the end, we replace estimated parallel lines with a virtual line of same slope with a mean offset. Corner is located at the intersection of the two virtual lines. The proposed method starts with the extraction of edge points based on a robust Canny detector as follows.

At the beginning, image data is converted to a floating point representation in order to minimize information loss in subsequent operations. Then, the image is smoothed with a Gaussian filter of optional size (noisier images will benefit from a larger filter size). We have used sizes from 1×1 pixels (no smoothing) to 9×9 pixels for different calibration datasets. The

¹http://www.cvlibs.net/datasets/kitti/eval_odometry.php

smoothed image is convolved with the Sobel filter, resulting with image gradients in the horizontal (G_x) and vertical (G_y) directions. From these two gradients, the magnitude and direction are computed as follows:

$$G = \sqrt{G_x^2 + G_y^2}, \quad \Theta = \text{atan2}(G_y, G_x). \quad (4)$$

Non-maximum pixels are suppressed from the gradient magnitude image, together with maximum pixels below some threshold. Checkerboard edges are usually the strongest edges in the scene, and proper threshold selection removes the majority of surrounding edges, speeding up the subsequent search for checkerboard corners. All the remaining edge pixels are localized with sub-pixel precision and paired with the corresponding direction. Then, pixels are set into one of the four direction quadrants: horizontal positive, horizontal negative, vertical positive, and vertical negative. Adjacent pixels belonging to the same direction quadrant are grouped into blobs. Four quadrants result in four possible blob types, each representing one of the square sides: top, bottom, left or right. Fig. 1 shows detected blobs for one of the boards of the KITTI calibration sequence 2011-10-03.

To initially locate the corners, we search for a “+” configuration of blobs, i.e., four different types of blobs connecting at a single point. In reality, blobs of interest often do not “touch”, but the ends of all four types reside in a small radius of several pixels. When such a configuration is found, we try to fit two horizontal parallel lines into the left and right blob. If we express the horizontal line with the following equation

$$Ax + y + C = 0, \quad (5)$$

then we can find the resulting parallel lines by minimizing the following criterion

$$\underset{A, C_1, C_2}{\text{argmin}} \sum_i \frac{(Ax_i + y_i + C_1)^2}{A^2 + 1} + \sum_j \frac{(Ax_j + y_j + C_2)^2}{A^2 + 1}, \quad (6)$$

for all inlier points (x_i, y_i) from the left blob i , and all inliers points (x_j, y_j) from the right blob j . We do an equivalent computation for vertical lines, except for switching the coordinate axes for numerical stability. Since the number of points is small, instead of using RANSAC, we test each possible hypothesis to determine the one with the maximum number of inliers. Minimal configuration for hypothesis generation consists of three points, where at least one point from each blob has to be selected. Although we do not employ edge direction information directly in the optimization formulation, we use it during the inlier selection. A point is selected as an inlier only if its distance from the line is below the distance threshold, and if the angle between its direction and line normal vector is below the angle threshold. Naturally, points in blob i are tested against the $[A, C_1]$ line, while points in blob j are tested against the $[A, C_2]$ line. Once $[A, C_1, C_2]$ is estimated, we choose a virtual $[A, (C_1 + C_2)/2]$ line as our estimate where the unique line would be in the ideal image. This is based on the assumption that the white square spills equally on both black squares in the local region of interest.

Table 1: KITTI reprojection error of our method, *libcdetect*, and OpenCV 4.2.0 `cornerSubPix()` function.

	libcdetect	OpenCV	ours
date	err. [px]	err. [px]	err. [px]
09-26	0.117022	0.116382	0.095147
09-28	0.128945	0.122802	0.099808
09-29	0.135855	0.128055	0.106468
09-30	0.135186	0.127309	0.106887
10-03	0.138393	0.129011	0.104743

Although overexposure is the most common phenomenon, the proposed approach is robust towards edge shifting in general, whether it is caused by overexposure, underexposure, bias in the edge detector, or even inaccurate printing. Furthermore, corner quality can be assessed a priori in the image space with different metrics, from simple number of inliers supporting the corner to total linear and angular variance of the fitted points re-weighted accordingly and fed into the calibration estimation with different weights. As a final comment, in reality it is often difficult or impossible to realize diffuse lighting; however, this problem can be alleviated by using matte surface calibration boards, since glossy surfaces have a strong reflection index and should be avoided if possible.

Regarding the results on the datasets, in Fig. 2 we compare the proposed corner detector (upper image, green) with original one (lower image, yellow) used in the default KITTI calibration. Magnified reprojection error reveals that the original corner detector is noisier than the proposed one, which also reflects on the reprojection error. Figures 3a and 3b show the proposed corner detector applied on an overexposed and underexposed KITTI calibration boards. Table 1 compares reprojection errors between the original, OpenCV 4.2.0 `cornerSubPix()` function, and the proposed corner subpixel refinement method (all used the same lens model). Error reduction for proposed method is notable on all the KITTI calibration images. Regarding the EuRoC dataset, in Fig. 3c we show the proposed corner detector applied to the EuRoC calibration board. The original image is darkened in order to make the overlay drawings more visible. Edge points of interest reside in the area between the smaller and larger green circle, which are user defined input parameters. Smaller circle is introduced due to erratic edge measurements in the corner vicinity.

2.2. Lens model

The most widely used projection model is the pinhole camera model which relates 3D coordinates of an object and its projection onto the image plane. Real lenses do not fit the pinhole camera model accurately enough, and therefore the pinhole model is usually accompanied with an appropriate lens distortion model. The most commonly used distortion model is the Conrady-Brown model proposed in [39] and [40, 4]. Distortion is modeled both in radial and decentering direction; therefore, also being known as the radial-tangential (rad-tan) distortion model, while the most common implementation is a simplified version presented in [4] (e.g., in OpenCV and Kalibr) as:

$$r = \sqrt{x^2 + y^2}, \quad (7)$$

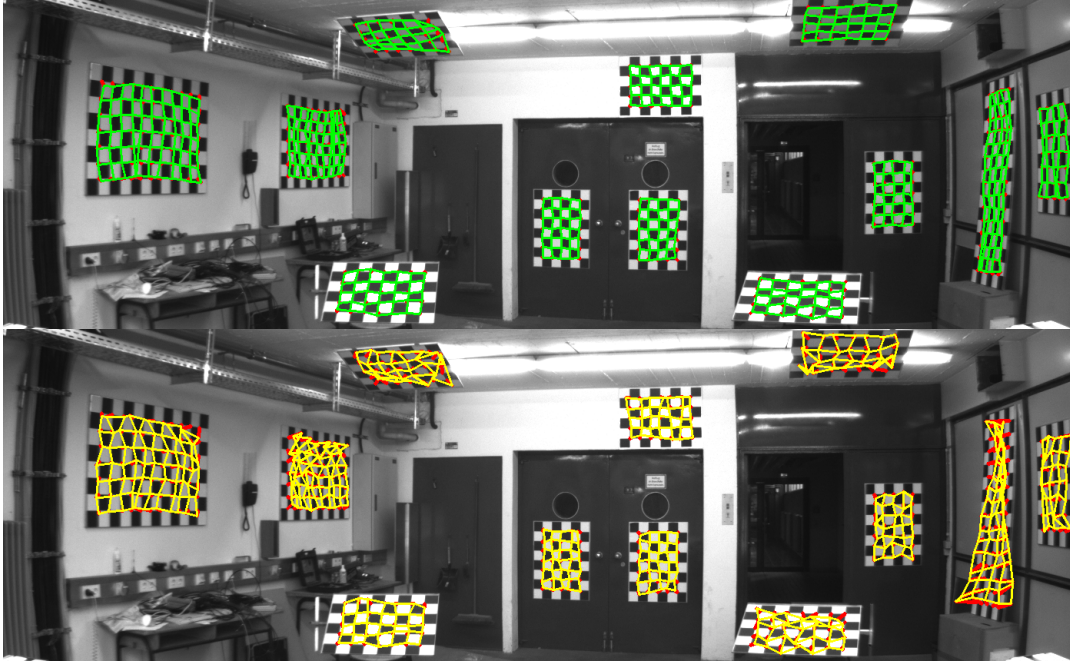
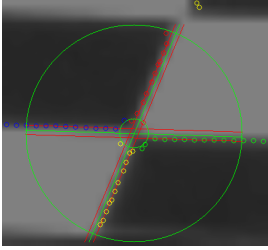
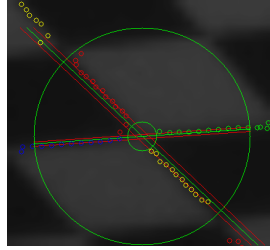


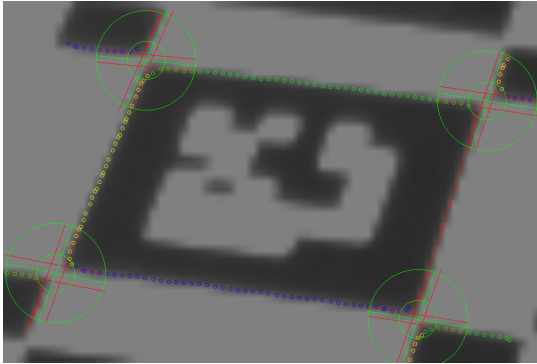
Figure 2: Reprojection error for the proposed line intersection method with dynamic board deformation (up, green) versus libcdetect (down, yellow) magnified 50 \times for the left frame 0 of KITTI calibration sequence 2011-10-03.



(a) Overexposed corner on a KITTI board



(b) Underexposed corner on a KITTI board



(c) Overexposed tag on a EuRoC board

Figure 3: Corner detection on challenging calibration images from the KITTI and EuRoC datasets (magnified 16 \times). Red dots – positive vertical edge points, yellow dots – negative vertical edge points, blue dots – positive horizontal edge points, green dots – negative horizontal edge points, parallel red lines – lines fitted to (6), green lines – final virtual lines whose intersection defines corner position. The area between the two green circles – a user defined parameter – a disc region around the corner in which edge segments can be considered linear (depends on the lens distortion).

$$x' = x \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + 2p_1 xy + p_2(r^2 + 2x^2), \quad (8)$$

$$y' = y \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + p_1(r^2 + 2y^2) + 2p_2 xy, \quad (9)$$

where distortion coefficients are

$$D = [k_1, k_2, p_1, p_2, k_3, k_4, k_5, k_6]. \quad (10)$$

Another popular model is the model proposed in [5], where it was stated that the pinhole model is inadequate for fish-eye lenses, because the 180 degree field of view cannot be projected to a finite plane. Fish-eye lenses are designed to obey a different projection – most commonly the equidistance projection, i.e., the projection on a sphere. However, to accurately model different types of lenses, authors proposed the following general model [5]:

$$\phi = \arctan \frac{y}{x}, \quad (11)$$

$$\theta = \arctan r, \quad (12)$$

$$\theta' = \theta(1 + k_1 \theta^2 + k_2 \theta^4 + k_3 \theta^6 + k_4 \theta^8), \quad (13)$$

$$x' = \theta' \cos(\phi), \quad (14)$$

$$y' = \theta' \sin(\phi), \quad (15)$$

with distortion coefficients

$$D = [k_1, k_2, k_3, k_4], \quad (16)$$

where we used the reduced version and slightly altered the equations to account for relying on the pinhole projection

Table 2: Mean reprojection errors for different distortion and board deformation models for the case of the EuRoC dataset. Static deformation accounts for board imperfections due to production, while dynamic deformation accounts that this deformation can change from one image to another.

distortion model	mean error [px]	board model	mean error [px]
rad-tan 4 + static	0.274963	rad-tan8	0.12805
rad-tan 5 + static	0.0806202	rad-tan8 + static	0.0319202
r-K-B + static	0.0378269	rad-tan8 + dynamic	0.10187
rad-tan 8 + static	0.0319202	rad-tan8 + static + dynamic	0.0289321

Table 3: Reprojection error comparison for the KITTI dataset for different lens distortion models and the dynamic board deformation model.

KITTI	rad-tan5	rad-tan8	rad-tan5 d	rad-tan8 d
date	err. [px]	err. [px]	err. [px]	err. [px]
09-26	0.095147	0.088288	0.050726	0.049022
09-28	0.099808	0.098393	0.053093	0.052172
09-29	0.106468	0.100128	0.054725	0.053285
09-30	0.104819	0.100385	0.049915	0.049915
10-03	0.104743	0.103905	0.052474	0.051650

model. We refer to this model as the Kannala-Brandt distortion. The reduced version contains only the radially symmetric distortion, while the full model also includes the asymmetric part. Nevertheless, in [5] it was found that using the reduced model is reasonable, since the asymmetric part is very flexible and can be subject to overfitting. Therefore, the most common implementations of the Kannala-Brandt distortion are the reduced versions: OpenCV (named *fish-eye camera model*) and Kalibr (named *equidistant distortion model*).

The EuRoC dataset provides calibration parameters for a simplified radial-tangential model with first four distortion parameters where parameters from k_3 to k_6 are set to zero (rad-tan4) and these parameters were obtained with the “Kalibr” package [29]. Note that in this section we include the static and dynamic board deformation models to account for board imperfections and leave only the lens model effects for consideration. However, details on the static and dynamic board deformation modeling are presented in Sections 2.3 and 2.4. We repeat the calibration procedure with the rad-tan4 model and analyze the reprojection error, where Fig. 4a shows the corresponding heat map. We can notice large reprojection errors at the image edges, which suggests presence of higher radial distortion than the selected model is able to capture. The next commonly used model in the community, rad-tan5, uses five distortion parameters, where parameters from k_4 to k_6 are set to zero. Fig. 4b shows the corresponding reprojection error heat map, from which we can notice that the error is significantly reduced; however, the remaining pattern with concentric circles suggests that this model is also not able to accurately represent the lens distortion. We also tested a reduced Kannala-Brandt (r-K-B) model, with a corresponding reprojection error heat map showed on Fig. 4c. Error distribution reveals that this model represent the lens distortion very accurately, with only slight inconsistency at image edges.

Finally, we tested a full radial-tangential model with all 8 parameters (rad-tan8), whereas the error heat map is shown in

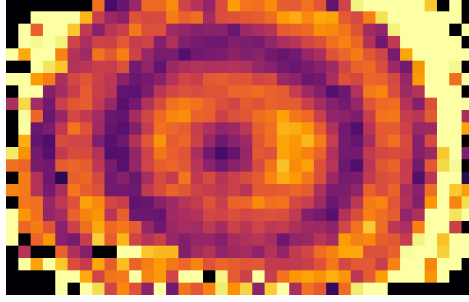
Fig. 4d. As we can notice, there is no recognizable pattern in the heat map and the remaining error mostly resembles uniform noise. Comparing heat maps of all tested models, this model seems to represent the lens the most accurately. Furthermore, Table 2 (two left columns) shows mean reprojection errors in pixels for all tested models. The lowest reprojection error was achieved by the rad-tan8 model; therefore, we decided to use this model in the odometry evaluation experiments.

Unlike the EuRoC dataset, the KITTI dataset comes with already pre-rectified images. However, the authors also provide raw images together with default calibration parameters, which reveal that the rad-tan5 distortion model was used for default calibration. Unfortunately, due to the single-shot approach, corner distribution across the image is very sparse, and we cannot analyze the error distribution with a heat map. For the same reason, model parameters on the majority of the image area are extrapolated, which is less precise compared to interpolated parameters. Given that, we analyze only the final reprojection error, where besides the default rad-tan5 model, we also test the full rad-tan8 model. In Table 3 we compare the reprojection errors of these models and we can see that the gain of the full rad-tan8 model is minor, and that simpler rad-tan5 model fits the lens almost as good as the full model.

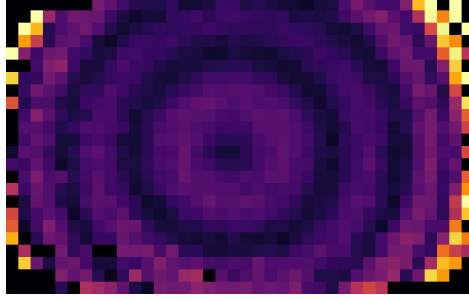
2.3. Static board deformation

Board deformation is very common in all calibration systems, but it is usually not taken into account. In this section we model the static deformation of the board that considers the fact that the calibration board is not necessarily an ideal plane, but rather an uneven surface, thus retaining its true shape as the board is moved from one pose to another. Each corner 3D position X is represented with a deviation from its ideal coordinate in all three directions. In order to prevent ambiguity with the board pose and retain absolute scale, 7 parameters need to remain fixed. We follow the approach from [17] and fix two corners, $\delta X_1 = (0, 0, 0)$, $\delta X_2 = (0, 0, 0)$, together with the last coordinate of a third corner $\delta X_3 = (\delta x, \delta y, 0)$.

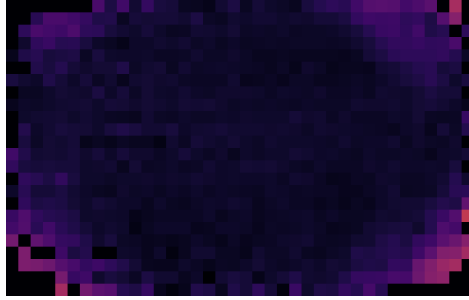
For the EuRoC dataset, the AprilTag board contains 36 AprilTags with 4 corners each, thus the state we estimate is augmented with $36 \cdot 4 \cdot 3 - 7 = 425$ additional unknowns. Fig. 5 depicts the estimated AprilTag board deformation estimated jointly with the calibration parameters. The largest perturbation is at the board center, with a deviation of 1.3 mm relative to the corners. Table 2 (two right columns) compares reprojection errors without any board deformation model (rad-tan 8) and with the static board deformation model included (rad-tan 8 + static). Note a significant decrease in the reprojection error



(a) Rad-tan 4 + static



(b) Rad-tan 5 + static



(c) r-K-B + static



(d) Rad-tan 8 + static

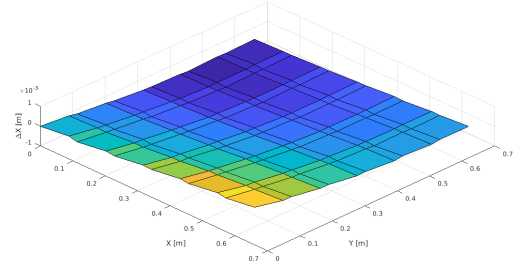


(e) Heatmap scale in pixels

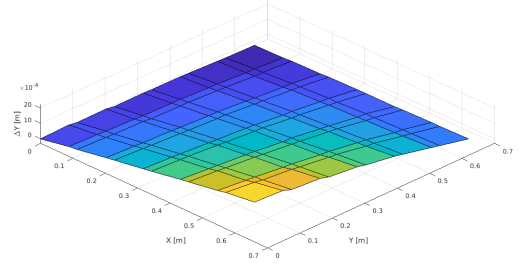
Figure 4: Reprojection error heat map for different distortion models and the EuRoC dataset.

when static board deformation is added. Namely, failing to observe the possibility that the calibration board is deformed may lead to a wasted search for more accurate distortion parameters, without any real chance for further reprojection error reduction.

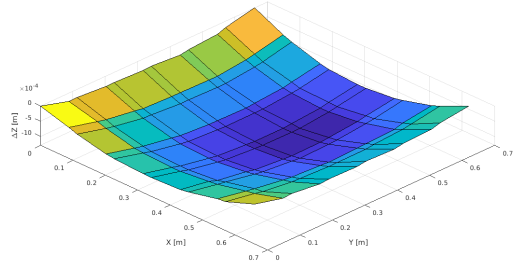
Since the KITTI dataset uses the one-shot approach with many different boards, each of them are captured only once,



(a) Deformation in X-direction



(b) Deformation in Y-direction



(c) Deformation in Z-direction

Figure 5: Static board deformation detected on the EuRoC AprilTag calibration board

thus we do not have enough board poses to constrain and accurately estimate high-parameter static board deformation for each board. Given that, in the case of KITTI, we do not apply the static deformation model.

2.4. Dynamic board deformation

The static board deformation model assumes that the shape of the board surface does not change from one pose to another. However, each calibration board has a specific flexibility coefficient and forms different shapes, depending on the pose relative to gravity direction, and distribution of support points. The larger the board, the greater the inflection. The dynamic board deformation model predicts that the board can have a different shape for each pose, but modeling a varying shape for each pose with completely free corner deviation, as in the case of the static deformation model, it would introduce too many unknowns and the problem would not be well-conditioned. Hence, a much simpler model should be used with low dimensional parametrization. In this paper, we adopt a sim-

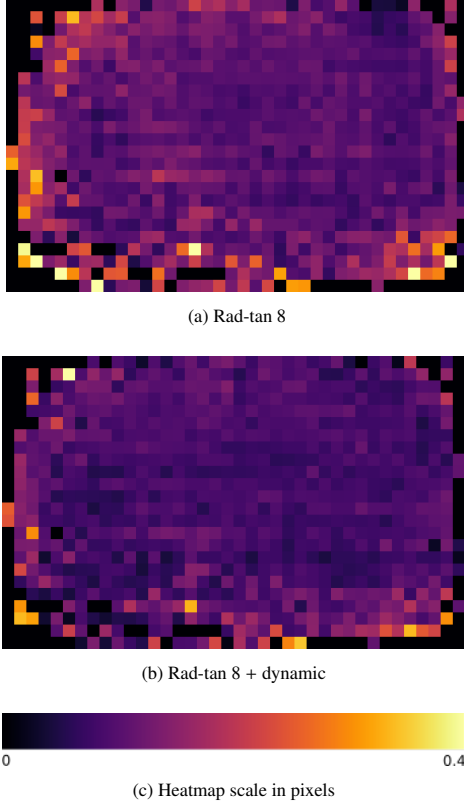


Figure 6: Heat map for different board models.

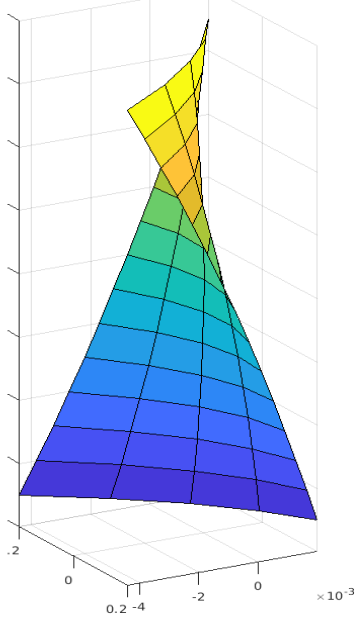


Figure 7: Dynamic board deformation detected on KITTI board no.11 (scale in meters).

ple paraboloid model proposed in [18]

$$dZ_i = a_j X_i^2 + b_j Y_i^2 + c_j X_i Y_i, \quad (17)$$

where X_i and Y_i are coordinates of the corner i relative to the board center, dZ_i is the resulting deviation of the corner i in the Z direction, and $[a_j, b_j, c_j]$ parameters of the board j paraboloid that we need to estimate. Note that this model introduces cor-

rections in the Z direction only, which violates the preservation of inter-corner distances. Although simple, this model is still capable of capturing deformation to some extent and reduce calibration errors. Unlike the static model, whose state size depends on the number of corners, the dynamic model augments the total state size with the number of board poses times three.

In Table 2 (two right columns), for the EuRoC dataset, we can notice reduction of the reprojection error by introducing the dynamic board deformation model. Moreover, by combining the static and dynamic deformation models, we obtained the best result. Furthermore, by comparing Fig. 6b and Fig. 6a we can see that the dynamic distortion model reduces the reprojection error uniformly across the image. On the other hand, from Fig. 4d we can see that the dynamic model was not able to capture the board deformation as accurately as the static model. Furthermore, an interesting observation is that model in Fig. 6a, which has no specific error pattern indicating a good lens model, compared to Fig. 4b, which does exhibit cocentric circles, has actually a larger total reprojection error – this further motivates a deeper analysis that goes beyond just looking at reprojection error values (cf. Section 3).

For the case of the KITTI dataset, in Table 3 we can see notable improvement with introduction of the dynamic board model. In the case of KITTI, word *dynamic* could be misleading. It is not a dynamic deformation of one board over the time, but rather a static deformation of each board in the single-shot image, represented by 3 parameters for our case. Fig. 7 shows deformation of the board no. 11 captured with a paraboloid [$a = -0.0190222, b = -0.000921217, c = 0.0227076$]. This shape is also visible from magnified reprojection errors shown in Fig. 2, bottom image, the second board from the right. Note in the upper image of the same figure, the nearly ideal plane for the same board in the case when the dynamic deformation is included.

3. Ex-post calibration analysis

While optimizing for the calibration parameters by minimizing (3), the main criterium is the reprojection error. However, low reprojection error does not guarantee necessarily the best possible set of parameters for a given application. To further explore possible improvements and gain valuable insights into the results, in this section we analyze: (i) calibration parameter uncertainty with respect to the corner location noise and (ii) reprojection error sensitivity concerning perturbed calibration parameters in the neighborhood of the optimum.

3.1. Uncertainty of estimated calibration parameters

In this subsection our goal is to estimate uncertainty of the calibration parameters. In the vein of [41] we start from estimating the noise of our measurement vector, where we treat corner coordinates x and y as two independent variables that are zero-mean and normally distributed. Presuming that we have a correct camera model, board deformation model, and absence of outliers, the measurement noise is then the only source of error, which seems a reasonable assumption by inspecting Fig. 4d.

The measurement noise is estimated as the sample variance of the residuals at the optimal calibration parameters value

$$\sigma^2 = \text{var}[e(p^*, z)]. \quad (18)$$

Note that σ^2 is a variance *factor* with respect to assumed a priori covariance that we set as identity [42, 43].

The gradient of the objective function at the optimal parameters value is zero

$$\nabla_p F(p, z) = 2J^T e(p, z) = 0, \quad (19)$$

where J is the Jacobian of the residuals $J = \frac{\partial e(p^*, z)}{\partial p}$. As in [41], to find the uncertainty of the calibration parameters, we perform a first-order Taylor expansion of the objective function

$$F(p + \delta p, z + \delta z) \approx F(p, z) + \nabla_p F(p, z) \delta p + \nabla_z F(p, z) \delta z^T (\dots). \quad (20)$$

By taking the gradient of (20)

$$\nabla_{\delta p} F(p + \delta p, z + \delta z) = 2J^T J \delta p + 2J^T \delta z = 0, \quad (21)$$

we obtain the following non-linear least squares solution

$$\delta p = -(J^T J)^{-1} J^T \delta z = M \delta z, \quad (22)$$

where all the Jacobians are evaluated at the optimal parameters value p^* . We find the covariance matrix of the parameters as the following transformation of the measurement variance

$$\Sigma = M \text{var}[z] M^T = \sigma^2 (J^T J)^{-1}. \quad (23)$$

In this analysis, we are especially interested in the variance of camera matrix parameters: focal length, principal point, and the baseline, since they largely affect the odometry trajectory. On the other hand, errors in distortion parameters are propagated to image reprojection errors in radial direction and a common technique for alleviating such errors is to carefully select uniformly distributed corners across the image. This way the mean of reprojection errors will tend to zero. Table 4 shows the standard deviations of camera matrix parameters for EuRoC and KITTI, with distortion models rad-tan5 and rad-tan8. We can see that the uncertainty depends on the selected model, i.e., models with fewer parameters are more confident. Since simpler models usually do not represent the lens perfectly, they tend to erroneously produce overconfident uncertainty estimates [41]. Furthermore, we can also notice difference in the values between the EuRoC and KITTI uncertainties. The reason behind this lies in the fact that for KITTI the calibration image includes only 12 distant targets, while the EuRoC dataset includes images of the target at both close and distant ranges. Using only distant range images produced intrinsic parameters uncertainty of about 0.3 pixels, while close range images yielded intrinsic parameters uncertainty of about 0.07 pixels. Nevertheless, uncertainty values in Table 4 can serve only as a good relative indicator, since most likely they are overconfident. Given that, in the next section we shall investigate how the reprojection error behaves with respect to varying parameter values.

3.2. Calibration parameters sensitivity analysis

Usually, the main criterium for computing calibration parameters is minimization of the total reprojection error. However, for the case of the KITTI dataset, parameters that we obtained using previously described models, as compared to parameters obtained with *libcbdetect*, yielded slightly more accurate odometry, but still below our expectations. This motivated us to conduct a sensitivity analysis of the reprojection error criterium by perturbing a certain calibration parameter in the neighborhood of the optimal value. For example, we take the value of f_x from the optimal solution, perturb its value with some small δ , and estimate all the other calibration parameters while holding the perturbed value of f_x fixed. We perturb f_x with a different δ value and repeat the process. In Fig. 8a we show the dependence of the reprojection error with respect to the varying f_x for the EuRoC and KITTI calibration setups. From the figure we can see that the EuRoC graph exhibits a strong minimum, while the KITTI graph is almost flat in the range of 20 pixels. Values 10 pixels away from the minimum point have an error increase of only 0.001 pixels (which is another indicator of overly optimistic deviation estimates in Table 4). Similar result can be seen for the principal point c_u in Fig. 8b. Although f_x and c_u , significantly influence odometry accuracy, the change in the reprojection error was insignificant. We concluded that the provided board configurations do not constrain enough the calibration for accurate estimation of all the parameters.

We also analyzed the scale which mostly depends on the estimated stereo baseline. Fig. 9 shows the reprojection error analysis with respect to varying baseline value and camera pitch angle. Again, we can notice that in this case the reprojection error for the EuRoC exhibits a strong minimum, while the KITTI curve is flat. However, there is a strong correlation between the baseline and the relative angle between the cameras. Forcing the baseline to a different value results in the change of the camera pitch angle around the y axis, while retaining almost the same reprojection error (note that local camera coordinate system is right-handed with the z axis pointing in the direction of the optical axis). Thus the system cannot disambiguate between the baseline and the pitch angle near the exact solution, since different triangulation setups lead to similar reprojection error.

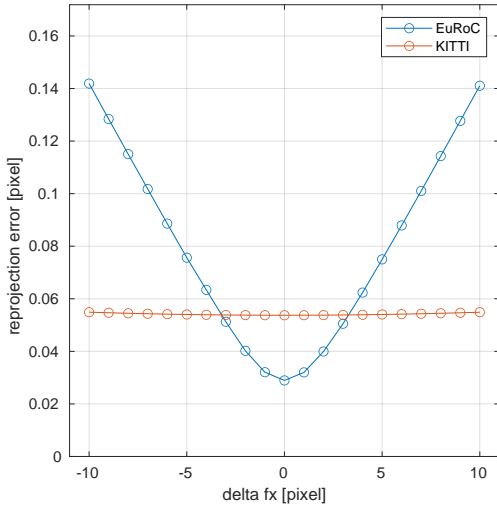
4. Experimental results

Throughout this paper, our main motivation for improving the calibration parameters was to obtain more accurate odometry trajectories. From this perspective, a lower reprojection error in the calibration procedure does not necessarily translate to more accurate odometry. Therefore, we conducted an extensive analysis of the obtained parameters by examining the odometry accuracy; specifically, the accuracy of four different visual odometries: SOFT-VO [1] and SOFT2² [2], ORB-SLAM2 [34], VINS-FUSION [35, 36] and VISO2 [37]. We

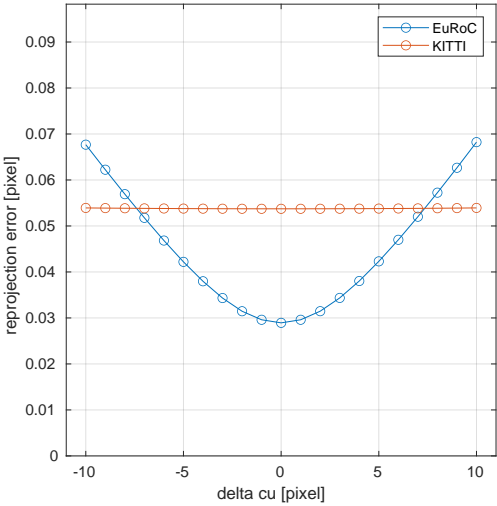
²SOFT2 is a 2D-2D odometry specifically designed for road vehicles and not appropriate for unconstrained motion datasets.

Table 4: Propagated uncertainty of computed parameters for different lens models and with static (s) or dynamic (d) deformation included.

	EuRoC rad-tan5-d		EuRoC rad-tan8-d		EuRoC rad-tan8-sd		KITTI rad-tan5-d		KITTI rad-tan8-d	
	value	σ	value	σ	value	σ	value	σ	value	σ
<i>meas</i>	0	0.085744	0	0.074173	0	0.021625	0	0.038284	0	0.037738
f_x	461.946	0.036387	462.492	0.039226	461.954	0.011544	975.444	1.01775	977.280	1.13623
f_y	460.653	0.036415	461.206	0.039276	461.899	0.012477	975.018	0.94241	976.323	1.00698
c_u	366.502	0.043077	366.341	0.037623	364.425	0.025426	685.753	2.55138	689.816	2.65130
c_v	249.262	0.043464	249.423	0.037792	248.111	0.031439	242.190	0.82127	241.946	0.86183
<i>base</i>	0.11001	0.000011	0.11002	0.000009	0.11014	0.000003	0.53285	0.00050	0.53346	0.00055



(a) Focal length f_x sensitivity test



(b) Principal point c_u sensitivity test

Figure 8: Sensitivity analysis of the reprojection error criterium with respect to varying calibration parameters. We can see a strong minimum for the EuRoC dataset and a rather flat curve for the KITTI dataset.

demonstrate the effectiveness of the obtained calibration parameters on the EuRoC and KITTI datasets. The EuRoC dataset is a visual-inertial dataset collected with a Micro Aerial Vehi-

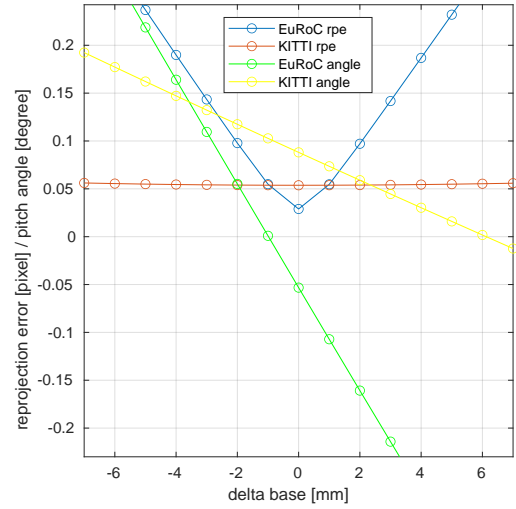


Figure 9: Stereo baseline sensitivity test showing the reprojection error with respect to varying baseline value and camera pitch angle.

cle (MAV). The dataset contains stereo images, synchronized IMU measurements, and accurate motion and structure ground-truth. In this paper we focus on stereo images, which are captured with an Aptina MT9V034 global shutter sensor, WVGA monochrome sensors, with a framerate of 20 Hz. The KITTI dataset was recorded in urban, rural, and highway scenarios by two stereo pairs (color and grayscale) set at a baseline of 0.54 m. Image rate was 10 Hz, with a resolution somewhat smaller than the original 1392×512 pixels due to rectification. High accuracy OXTS3003 GPS/IMU unit was used for recording the ground truth trajectory. Even though we have focused on the EuRoC and KITTI datasets, the proposed process can be applied on any camera setup. The calibration files containing all the parameters for both datasets are publicly available ³.

4.1. EuRoC dataset

For the EuRoC dataset, somewhat surprisingly, the model with the lowest reprojection error of 0.0289321 pixels, the rad-tan8 model with static and dynamic deformation, achieved the second best result. The most accurate result was achieved with the rad-tan8 model and dynamic board deformation, although

³<https://bitbucket.org/unizg-fer-lamor/kittical/>

Table 5: Visual odometry results with default and custom calibration for the EuRoC dataset (absolute trajectory error in meters, rt – rad-tan model, sd – static and dynamic deformation models, d – dynamic deformation model). Note that [*] was not added to the sum for a fair comparison.

	SOFT-VO			VINS-FUSION			ORB-SLAM2			VISO2		
	def.	rt8-sd	rt8-d	def.	rt8-sd	rt8-d	def.	rt8-sd	rt8-d	def.	rt8-sd	rt8-d
MH 01	0.032	0.053	0.022	0.126	0.124	0.104	0.037	0.041	0.039	0.180	0.180	0.126
MH 02	0.052	0.048	0.037	0.172	0.062	0.054	0.048	0.054	0.053	0.221	0.190	0.195
MH 03	0.150	0.103	0.120	0.202	0.129	0.147	0.039	0.045	0.043	0.209	0.184	0.159
MH 04	0.196	0.163	0.133	0.289	0.289	0.217	0.117	0.077	0.053	0.628	0.538	0.519
MH 05	0.107	0.113	0.126	0.266	0.234	0.201	0.112	0.060	0.066	0.327	0.293	0.237
V1 01	0.039	0.053	0.042	0.563	0.536	0.534	0.087	0.088	0.087	0.070	0.088	0.082
V1 02	0.070	0.062	0.063	0.234	0.156	0.173	0.081	0.151	0.119	0.176	0.098	0.076
V1 03	0.119	0.108	0.078	x	x	x	0.339	0.379	0.357	0.526	0.441	0.391
V2 01	0.107	0.091	0.126	0.225	0.136	0.145	0.067	0.071	0.067	x	0.192*	0.189*
V2 02	0.118	0.137	0.148	0.206	0.146	0.148	0.108	0.101	0.110	x	0.478*	0.302*
SUM	0.990	0.931	0.895	2.283	1.812	1.723	1.035	1.067	0.994	2.337	2.012	1.785

its reprojection error of 0.10187 is significantly higher. We hypothesize that many parameters of the static board deformation model lead to a slight overfit. We also corrected the extrinsic rotation of the right camera for 0.24 deg around the x -axis for V2 tracks in order to align stereo epipolar lines. This misalignment exists only in V2 tracks and was probably caused by a slight bump to the VI-sensor prior to recording.

In Table 5 we show the results for SOFT-VO, VISO2, VINS-FUSION and ORB-SLAM2. However, making large progress on EuRoC is not easy since default parameters are pretty accurate. Odometry algorithms usually do not exploit 0.03 pixel corner detection uncertainty, and reprojection error of 0.2 pixel obtained with the Kalibr package is good enough if the camera matrix parameters are accurately estimated. Still, we managed to improve the odometry on most tracks (note that V1 03 track is classified as *difficult*, with aggressive movements, motion blur, and sudden brightness change, which present the biggest obstacle towards good accuracy for this track).

The results for the SOFT-VO and VISO2 are probably the most indicative, since these are pure frame-to-frame odometries, and therefore very sensitive to calibration parameters. Another advantage of SOFT-VO and VISO2 in this experiment is their deterministic nature – each run produces exactly the same result. That raises probability that the difference in odometry error is only caused by calibration parameters change and we had to run experiments for SOFT-VO and VISO2 only once.

A complete opposite is ORB-SLAM2, where calibration errors can get averaged out, due to dense inter-frame connections among features. This is especially the case with the EuRoC dataset, where the whole trajectory is in a single room, producing numerous keyframe links for the local bundle adjustment, which prevents odometry drift, and it is difficult to make further improvement just with new calibration. We did not test the newer ORB-SLAM3 [44] because fusion with IMU would additionally cloud the effect of different calibration parameters. Results of ORB-SLAM2 vary significantly between different runs, so we adopted a metric from the original paper [34] – each track is ran five times and the mean value is taken as the final result. Given that, we managed to only slightly improve

ORB-SLAM2 results on the EuRoC dataset.

Results for VINS-FUSION vary slightly between different runs, despite the single-threaded mode being selected in the configuration, but are often very similar with few occasional outliers. Given that, the final results in Table 5 for VINS-FUSION are the median of three runs. At this point, we also need to discuss the difference between default VINS-FUSION results in Table 5 and the ones presented in [45], Table I, stereo. Therein, authors used custom calibration parameters, where the principal point for both the left and right camera was different than the default EuRoC calibration. However, *bodyToCam0* and *bodyToCam1* transforms were used from the original EuRoC calibration files, although different principal point positions in the custom calibration file also define a new camera coordinate systems, and consequently new relative transform from one to the other. This new transform also has to be updated in the stereo calibration file. Given that, in our experiments we used the original EuRoC calibration files in Table 5, which produced notably better results than the ones presented in [45]. Also, note that our custom results for Machine Hall tracks are even better than the original camera and IMU combinations.

4.2. KITTI dataset

Since we concluded that the KITTI board configurations do not constrain calibration enough for targeted estimation of all the parameters, we decided to change the objective function and search for parameters with the focal length f_x , principal point (c_u, c_v) , and baseline that minimize the KITTI translational error (there is a strong correlation between f_x and f_y and refinement in one is instantly reflected in the other). We employed grid search around the initial minimum such that the four values were fixed during the calibration process, while the other parameters varied to minimize the reprojection error. Then, the odometry with final calibration parameters was evaluated for the translational error on the training sequences. The best parameters were usually found within the neighborhood of 10 pixels. While the reprojection error increased for 1%, odometry error was reduced by 30% in translation and 40% in rotation. Note that in the calibration process, the focal length and prin-

Algorithm 1 KITTI calibration parameters search

Require: Initial calibration parameters for grid search $(f_x^0, c_u^0, c_v^0, b^0)$ and initial odometry error $init_odom_err$.

Ensure: Final calibration parameters $final_cal_params$.

```
1: Define search grid for parameters:
    $search\_grid \leftarrow (\Delta f_x, \Delta c_u, \Delta c_v, \Delta b)$ 
2:  $curr\_odom\_err \leftarrow init\_odom\_err$ 
3:  $converged = false$ 
4: while ( $\neg converged$ ) do
5:   Calibrate for other parameters:
      $cal\_params\_f_x = full\_cal(f_x^0 \pm \Delta f_x(i), c_u^0, c_v^0, b^0)$ 
6:   Run odometry and compute error:
      $err\_f_x \leftarrow run\_odom(calib\_params\_f_x)$ 
7:   Repeat steps 4 and 5 for  $\Delta c_u(i), \Delta c_v(i), \Delta b(i)$ 
8:   ...
9:    $cal\_params = \{cal\_params\_f_x, cal\_params\_c_u,$ 
      $cal\_params\_c_v, cal\_params\_b\}$ 
10:   $odom\_error \leftarrow \{err\_f_x, err\_c_u, err\_c_v, err\_b\}$ 
11:   $min\_odom\_err \leftarrow \min(odom\_error)$ 
12:  if  $min\_odom\_err < curr\_odom\_err$  then
13:    Select calibration parameters from the min error run:
14:     $opt\_cal\_params \leftarrow cal\_params(@min\_odom\_err)$ 
15:    Extract subset for grid search:
16:     $(f_x^0, c_u^0, c_v^0, b^0) \leftarrow get\_fx\_cuv\_b(opt\_cal\_params)$ 
17:     $curr\_odom\_err = min\_odom\_err$ 
18:  else
19:     $converged = true$ 
20:  end if
21: end while
22:  $final\_cal\_params \leftarrow opt\_cal\_params$ 
```

cial point were fixed for the left camera only, while the corresponding parameters in the right camera are highly correlated via boards and they automatically change with the probed left camera values. Also, the system cannot disambiguate between the baseline and the pitch angle near the exact solution, since different triangulation setups lead to similar reprojection error. Consequently, exact baseline is hard to assess with odometry, since the trajectory scale may also depend on the pitch angle. In the end, a total number of 4 parameters is refined via grid search: f_x and (c_u, c_v) of the left camera, and the baseline. In Algorithm 1 we provide pseudocode of this algorithm. Note that search is performed twice independently, once with single calibration image 10-03, which corresponds to sequences 00-02 (recorded on October 3), and once with single calibration image 09-30, which corresponds to sequences 04-10 (recorded in September 30).

Table 6 compares results for SOFT2, VINS-FUSION, ORB-SLAM2, and VISO2 obtained with default and custom calibration parameters. For custom calibration, we seeded the radtan8-d model at $[f_x = 976, c_u = 690, c_v = 249, b = 0.535]$. In order to emphasize the effect of calibration parameters on the odometry trajectory, loop closing features of VINS-FUSION and ORB-SLAM2 were not used. As we can see from the table both translational and rotational error is considerably reduced

for all the tested algorithms. Although we used a more complex lens model and board deformation, for KITTI we obtained similar results to those presented in [38]. There are two possible reasons, one is that due to the board positions calibration is not able to exploit better the more complex models, and the other is applying grid search for a specific odometry algorithm. Each odometry algorithm has its own specific “biases”, and pushing calibration parameters towards more accurate trajectory of one algorithm can cause decrease in accuracy of the other algorithms (herein we optimized for ORB-SLAM2).

5. Conclusion

In this paper we have presented an enhanced approach to camera setup calibration and parameter analysis. We focused on the EuRoC and KITTI datasets, inspired by their significance in the visual odometry community and motivated by our experience of achieving the highest localization score during the EuRoC competition [1] and of currently holding the highest odometry rank on the KITTI score board [2]. The proposed approach consists of an improved corner detection procedure, focusing on line intersections for sub-pixel localization accuracy. We analyzed Conrady-Brown and Kannala-Brandt lens models and included static and dynamic board deformation in the calibration pipeline. The whole procedure yielded smaller reprojection errors on both datasets. Additionally, we performed an ex-post calibration analysis by investigating uncertainty of calibration parameters and the sensitivity of the reprojection error criterion. For the KITTI dataset specifically, since the board configurations do not constrain enough all the camera parameters, we use a grid search procedure based on the accuracy of the translational error in the KITTI score metric. Finally, we assess the accuracy of different visual odometry algorithms using the newly obtained calibration parameters: SOFT [1, 2], ORB-SLAM2 [34], VINS-FUSION [35, 36] and VISO2 [37]. All the algorithms exhibited improved odometry accuracy on most of the EuRoC and KITTI tracks.

Acknowledgment

This research has been supported by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DAT-ACROSS).

References

- [1] I. Cvišić, J. Česić, I. Marković, I. Petrović, SOFT-SLAM: Computationally Efficient Stereo Visual SLAM for Autonomous UAVs, *Journal of Field Robotics* 35 (4) (2018) 578–595.
- [2] I. Cvišić, I. Marković, I. Petrović, SOFT2: Visual Stereo Odometry for Ground Vehicles based on a Point-to-Epipolar-Line Metric, *IEEE Transactions on Robotics*, in review.
- [3] C. Geyer, K. Daniilidis, A unifying theory for central panoramic systems and practical implications, in: *European Conference on Computer Vision (ECCV)*, 2000, pp. 445–461.
- [4] D. C. Brown, Close-range camera calibration (1971).
- [5] J. Kannala, S. S. Brandt, A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (8) (2006) 1335–1340.

Table 6: Visual odometry results with default and custom calibration for 10 KITTI dataset train sequences. We used the KITTI metric for the translational tr [%] and rotational rot [deg/100 meters] errors.

Seq.	SOFT2				VINS-FUSION				ORB-SLAM2				VISO2			
	default		custom		default		custom		default		custom		default		custom	
	tr	rot	tr	rot	tr	rot	tr	rot	tr	rot	tr	rot	tr	rot	tr	rot
00	0.68	0.28	0.54	0.18	1.29	0.63	0.80	0.33	0.88	0.31	0.55	0.18	1.53	0.64	0.80	0.31
01	1.07	0.16	0.84	0.10	2.29	0.42	1.84	0.22	1.44	0.19	1.39	0.28	3.80	0.70	2.14	0.22
02	0.73	0.21	0.56	0.14	1.37	0.50	1.02	0.38	0.77	0.28	0.55	0.15	1.61	0.52	1.04	0.27
04	0.55	0.07	0.30	0.07	1.35	0.72	1.89	0.64	0.46	0.19	0.37	0.23	1.29	0.51	1.61	0.30
05	0.61	0.24	0.40	0.17	1.49	0.74	1.31	0.54	0.62	0.26	0.54	0.16	1.25	0.66	0.78	0.32
06	0.60	0.23	0.31	0.21	1.35	0.71	0.96	0.35	0.80	0.25	0.31	0.12	0.79	0.51	0.94	0.33
07	0.50	0.33	0.30	0.16	1.21	0.90	0.53	0.38	0.89	0.50	0.36	0.20	1.46	1.13	0.78	0.54
08	1.00	0.28	0.80	0.16	1.83	0.72	1.41	0.46	1.03	0.31	0.83	0.18	1.62	0.66	1.09	0.41
09	0.75	0.18	0.59	0.08	1.82	0.53	1.60	0.36	0.86	0.25	0.80	0.15	0.84	0.64	1.29	0.47
10	0.65	0.24	0.58	0.14	2.64	1.01	1.88	0.61	0.62	0.29	0.70	0.21	1.29	0.64	1.44	0.38
avg	0.76	0.24	0.55	0.13	1.57	0.63	1.18	0.41	0.87	0.29	0.65	0.18	1.62	0.63	1.03	0.34

- [6] T. Schops, V. Larsson, M. Pollefeys, T. Sattler, Why Having 10,000 Parameters in Your Camera Model Is Better Than Twelve, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2020, pp. 2532–2541.
- [7] A. Richardson, J. Strom, E. Olson, AprilCal: Assisted and repeatable camera calibration, in: IEEE International Conference on Intelligent Robots and Systems, 2013, pp. 1814–1821.
- [8] S. Peng, P. Sturm, Calibration wizard: A guidance system for camera calibration based on modelling geometric and corner uncertainty, in: IEEE International Conference on Computer Vision (ICCV), 2019, pp. 1497–1505.
- [9] J.-Y. Bouguet, Camera Calibration Toolbox for Matlab. URL http://www.vision.caltech.edu/bouguetj/calib_doc/
- [10] Z. Zhang, A flexible new technique for camera calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (11) (2000) 1330–1334.
- [11] Q. T. Luong, O. D. Faugeras, Self-Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices, International Journal of Computer Vision 22 (3) (1997) 261–289.
- [12] M. Pollefeys, R. Koch, L. Van Gool, Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Intrinsic Camera Parameters, International Journal of Computer Vision 32 (1) (1999) 7–25.
- [13] J. Levinson, S. Thrun, Automatic Online Calibration of Cameras and Lasers, in: Robotics: Science and Systems (RSS), 2013.
- [14] A. Geiger, F. Moosmann, O. Car, B. Schuster, Automatic camera and range sensor calibration using a single shot., in: IEEE Conference on Robotics and Automation (ICRA), 2012, pp. 3936–3943.
- [15] C. Fang, S. Ding, Z. Dong, H. Li, S. Zhu, P. Tan, Single-Shot is Enough: Panoramic Infrastructure Based Calibration of Multiple Cameras and 3D LiDARs (2021). [arXiv:2103.12941](https://arxiv.org/abs/2103.12941).
- [16] A. Albarelli, E. Rodolà, A. Torsello, Robust camera calibration using inaccurate targets, in: British Machine Vision Conference, BMVC 2010 - Proceedings, 2010.
- [17] K. H. Strobl, G. Hirzinger, More accurate pinhole camera calibration with imperfect planar target, in: Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 1068–1075.
- [18] A. Hagemann, M. Knorr, C. Stiller, Modeling dynamic target deformation in camera calibration (2021). [arXiv:2110.07322](https://arxiv.org/abs/2110.07322).
- [19] N. Keivan, G. Sibley, Online SLAM with any-time self-calibration and automatic change detection, in: IEEE International Conference on Robotics and Automation (ICRA), 2015, pp. 5775–5782. [arXiv:1411.1372](https://arxiv.org/abs/1411.1372).
- [20] E. Rehder, C. Kinzig, P. Bender, M. Lauer, Online stereo camera calibration from scratch, in: IEEE Intelligent Vehicles Symposium, 2017, pp. 1694–1699.
- [21] P. Furgale, J. Rehder, R. Siegwart, Unified temporal and spatial calibration for multi-sensor systems, in: IEEE International Conference on Intelligent Robots and Systems (IROS), IEEE, 2013, pp. 1280–1286.
- [22] J. Maye, P. Furgale, R. Siegwart, Self-supervised calibration for robotic systems, in: IEEE Intelligent Vehicles Symposium, IEEE, 2013, pp. 473–480.
- [23] P. Furgale, C. H. Tong, T. D. Barfoot, G. Sibley, Continuous-time batch trajectory estimation using temporal basis functions, International Journal of Robotics Research 34 (14) (2015) 1688–1710.
- [24] A. Datta, J. S. Kim, T. Kanade, Accurate camera calibration using iterative refinement of control points, 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops 2009 2009 (October) (2009) 1201–1208.
- [25] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
- [26] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: The KITTI dataset, The International Journal of Robotics Research 32 (11) (2013) 1231–1237.
- [27] J.-L. Blanco-Claraco, F.-Á. Moreno-Duenas, J. González-Jiménez, The Malaga Urban Dataset : High-rate Stereo and Lidars in a realistic urban scenario, The International Journal of Robotics Research 33 (2) (2013) 207–214.
- [28] J. Xie, M. Kiefel, M. T. Sun, A. Geiger, Semantic Instance Annotation of Street Scenes by 3D to 2D Label Transfer, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2016-December (2016) 3688–3697.
- [29] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, R. Y. Siegwart, The EuRoC micro aerial vehicle datasets, The International Journal of Robotics Research 35 (10) (2016) 1157–1163.
- [30] W. Maddern, G. Pascoe, C. Linegar, P. Newman, 1 year, 1000 km: The Oxford RobotCar dataset, International Journal of Robotics Research 36 (1) (2017) 3–15.
- [31] A. Z. Zhu, D. Thakur, T. Özarslan, B. Pfrommer, V. Kumar, K. Daniilidis, The Multi Vehicle Stereo Event Camera Dataset: An Event Camera Dataset for 3D Perception, IEEE Robotics and Automation Letters 3 (3) (2018) 2032–2039.
- [32] M. Gehrig, W. Aarents, D. Gehrig, D. Scaramuzza, DSEC: A Stereo Event Camera Dataset for Driving Scenarios, IEEE Robotics and Automation Letters (2021) 1–8.
- [33] I. Krešo, S. Šegvić, Improving the egomotion estimation by correcting the calibration bias, in: International Conference on Computer Vision Theory and Applications, 2015, pp. 347–356.
- [34] R. Mur-Artal, J. D. Tardos, ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras (2016). [arXiv:1610.06475](https://arxiv.org/abs/1610.06475).
- [35] T. Qin, P. Li, S. Shen, VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator, IEEE Transactions on Robotics 34 (4) (2018) 1004–1020.
- [36] T. Qin, S. Shen, Online Temporal Calibration for Monocular Visual-Inertial Systems, in: IEEE International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 3662–3669.
- [37] A. Geiger, J. Ziegler, C. Stiller, StereoScan: Dense 3D Reconstruction in

- Real-time, in: Intelligent Vehicles Symposium (IV), 2011.
- [38] I. Cvisic, I. Markovic, I. Petrovic, Recalibrating the KITTI dataset camera setup for improved odometry accuracy, in: European Conference on Mobile Robots (ECMR), 2021. [arXiv:2109.03462](#),
 - [39] C. A. E., Decentred lens-systems, Monthly Notices of the Royal Astronomical Society (79(5)) (1919) 384–390.
 - [40] B. D. C., Decentering distortion of lenses, Photogrammetric Engineering (32) (1966) 444–462.
 - [41] D. Kogan, mrcal (2022).
URL <http://mrcal.secretsauce.net>
 - [42] W. Förstner, Reliability analysis of parameter estimation in linear models with applications to mensuration problems in computer vision, Computer Vision, Graphics and Image Processing (40) (1987) 273–310.
 - [43] K.-R. Koch, Parameter Estimation in Linear Models, 1999.
 - [44] C. Campos, R. Elvira, J. J. Rodriguez, J. M. Montiel, J. D. Tardos, ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM, IEEE Transactions on Robotics 37 (6) (2021) 1874–1890.
 - [45] T. Qin, S. Cao, J. Pan, S. Shen, A General Optimization-based Framework for Global Pose Estimation with Multiple Sensors (2019). [arXiv:1901.03642](#).