Computer Vision and Image Understanding
journal homepage: www.elsevier.com

MC-Calib: a generic and robust calibration toolbox for multi-camera systems

Francois Rameau[a], Jinsun Park[b], Oleksandr Bailo[c], In So Kweon[a,**]

[a]*KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea*
[b]*Pusan National University (PNU), Busan 46241, Republic of Korea*
[c]*Independent researcher*

ABSTRACT

In this paper, we present MC-Calib, a novel and robust toolbox dedicated to the calibration of complex synchronized multi-camera systems using an arbitrary number of fiducial marker-based patterns. Calibration results are obtained via successive stages of refinement to reliably estimate both the poses of the calibration boards and cameras in the system. Our method is not constrained by the number of cameras, their overlapping field-of-view (FoV), or the number of calibration patterns used. Moreover, neither prior information about the camera system nor the positions of the checkerboards are required. As a result, minimal user interaction is needed to achieve an accurate and robust calibration which makes this toolbox accessible even with limited computer vision expertise. In this work, we put a strong emphasis on the versatility and the robustness of our technique. Specifically, the hierarchical nature of our strategy allows to reliably calibrate complex vision systems even under the presence of noisy measurements. Additionally, we propose a new strategy for best-suited image selection and initial parameters estimation dedicated to non-overlapping FoV cameras. Finally, our calibration toolbox is compatible with both, perspective and fisheye cameras. Our solution has been validated on a large number of real and synthetic sequences including monocular, stereo, multiple overlapping cameras, non-overlapping cameras, and converging camera systems. Project page: `https://github.com/rameau-fr/MC-Calib`

1. Introduction

The recent years have seen a rapid increase in the demand for polydioptric (multi-camera setup) vision systems in multiple fields such as autonomous vehicle navigation (Heng et al., 2019), human reconstruction (Alexiadis et al., 2016), indoor robotics applications (Urban et al., 2016; Kuo et al., 2020) and video surveillance (Rameau et al., 2014). These systems are particularly desirable since they allow covering a large field-of-view (FoV) and computing metric scale 3D information from a scene. Despite advantages, these systems remain complex to deploy in practice due to their tedious calibration and the absence of efficient and versatile publicly available calibration toolboxes.

While significant efforts have been invested towards robust and effective software dedicated to Structure from Motion (SfM) (Schönberger and Frahm, 2016; Moulon et al., 2016; Wu et al., 2011) and Simultaneous Localization and Mapping (SLAM) (Mur-Artal and Tardós, 2017; Rosinol et al., 2020; Qin et al., 2018), the development of novel calibration toolboxes for complex vision systems attracted significantly less attention. As a result, most existing software for camera calibration focuses on monocular and stereo systems (Bouguet, 2004; Mei and Rives, 2007; Scaramuzza et al., 2006) but does not consider the problem of a multi-camera rig.

The relevance of our work can be understood in the context where existing calibration frameworks dedicated to polydioptric systems are often designed to deal with specific and restricted setups. For instance, a given and limited number of cameras (Bouguet, 2004), an overlapping FoV (Rehder et al., 2016), prior knowledge on the intrinsic parameters (Lébraly13 et al., 2010), external vision systems (Zhao et al., 2018), mirror (Kumar et al., 2008; Lébraly et al., 2010), limited motion (Liu et al., 2016) or pre-computed reconstruction of the environment (Lin et al., 2020; Ataer-Cansizoglu et al., 2014) are

**Corresponding author: Tel.: +82-42-350-5465;
e-mail: `iskweon77@kaist.ac.kr` (In So Kweon)

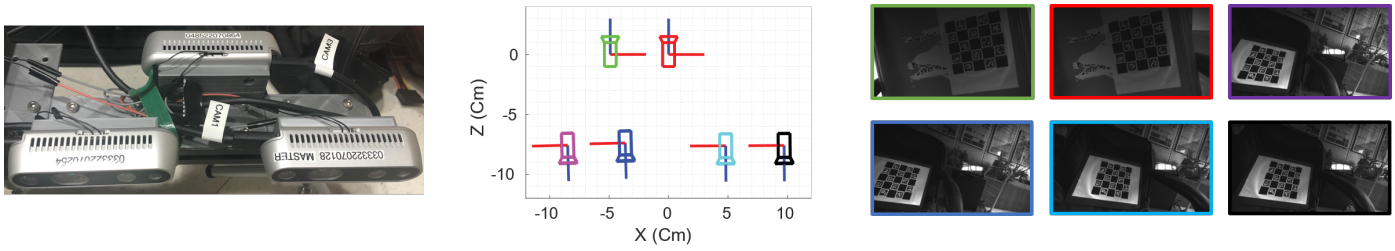


Fig. 1. Representative calibration result obtained with our calibration pipeline. (left) Camera rig composed of 3 Intel RealSense (Keselman et al., 2017) cameras where the 6 infrared cameras are being calibrated, (middle) calibration result, (right) image samples from each camera.

often required. Moreover, most of these available toolboxes are not currently compatible with converging camera systems (Yu et al., 2020) which are needed for a full 3D object or human body reconstruction (Alexiadis et al., 2016). In addition, these calibration techniques have often been designed either for perspective or fisheye cameras but rarely for both.

In this work, we propose a versatile and user-friendly toolbox compatible with any – perspective, fisheye, and hybrid – multi-camera configurations: overlapping, non-overlapping, and converging multi-camera systems. Our solution relies on fiducial markers (Munoz-Salinas, 2012) to jointly calibrate the intrinsic and extrinsic parameters of the cameras from a sequence of synchronized images without any manual interaction. The proposed approach is neither restricted by the number of cameras in the system nor their overlapping FoV. Moreover, an arbitrary number of checkerboards and 3D calibration objects (composed of a set of planar calibration targets) can be utilized to perform the estimation of the cameras' parameters. Unlike existing approaches (Bouguet, 2004; Forbes et al., 2002), no prior information regarding these 3D objects is needed beforehand since the geometry of the objects is computed automatically. An example of a calibrated system is shown in Fig. 1.

For this complex task, we develop new techniques to drastically improve the versatility, robustness, and effectiveness of the multi-camera calibration. Particularly, we design novel solutions to improve the stability and accuracy of non-overlapping camera calibration via the best image selection, bootstrapped initialization, and successive non-linear refinements.

To sum up our calibration process, first, all the observed checkerboards are used to calibrate the intrinsic parameters of each camera. After, if multiple calibration boards are visible in a single image, their relative poses are computed and stored in a graph. Under the common assumption that the boards are static (or rigidly attached), this graph is used to combine all the boards sharing covisibility (*i.e.*, visible simultaneously in an image) to form 3D calibration *objects*. Following the initial estimation of the 3D geometry of these objects, their 3D structures (poses between boards) are refined via a bundle adjustment strategy.

After estimating all the 3D calibration objects in the scene, the relative poses of each camera - w.r.t the 3D objects - are computed. This initial estimation is then used to compute the extrinsic parameters between each camera pair sharing an overlapping FoV in the vision system. To ensure an accurate estimation, these inter-camera poses are refined via a non-linear refinement process. From these camera pairs, we can form a graph representing all the combinations of camera pairs. The

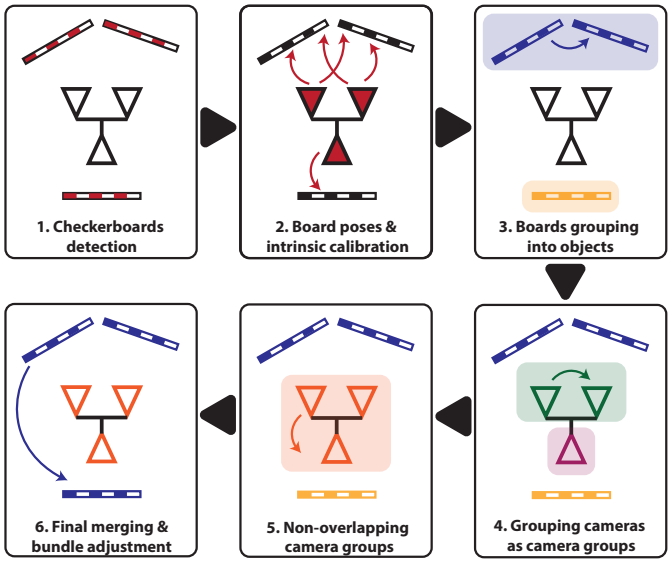


Fig. 2. MC-Calib pipeline.

cameras in the graph forming a single connected component are then merged into groups and their relative pose w.r.t a reference camera is computed. In this paper, we simply call these groups of cameras: *camera groups*.

If multiple *camera groups* are available, the estimation of the poses between them is performed. This situation typically occurs when no overlapping FoV exists between the camera groups. For this, we employ a well-known linear hand-eye non-overlapping calibration technique (Tsai et al., 1989) between each pair of camera groups. The resulting initial poses between camera groups are used to estimate the relative pose between all the cameras w.r.t an arbitrarily selected reference camera. The final stage of our calibration is the joint non-linear refinement of all the parameters (*i.e.*, inter-camera poses, inter-board poses, and intrinsic parameters) via a bundle adjustment. This entire calibration process is summarized in Fig. 2.

2. Background

This section briefly explains cameras' intrinsic and extrinsic parameters and introduces the notations used in the paper.

2.1. Camera parameters

The main goal of our calibration pipeline is to accurately compute both the intrinsic and extrinsic parameters of a set of

cameras rigidly attached together. The intrinsic parameters of a camera refer to the set of parameters mapping the projection of the 3D world onto the image plane. Assuming no geometric distortion induced by the lens, the geometry of the sensor can be approximated by the pinhole model. With this model, the perspective projection is parametrized by the camera matrix

$$\mathbf{K} = \begin{bmatrix} f & s & u_0 \\ 0 & \lambda f & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (1)$$

encapsulating five parameters modeling the projection, namely, the focal length f, the pixels' aspect ratio λ, the skew parameter s (representing the pixel non-orthogonality), and the position of the principal point in the image $pp = (u_0, v_0)^T$ (orthogonal projection of the camera center onto the image plane). Note that we assume a zero skew factor in our calibration process. A 3D point $\mathbf{P} = (X, Y, Z)^T$, expressed in the camera referential, can be projected onto the image at a pixel location $\mathbf{p} = (x, y, 1)^T$ (homogeneous notation) as follow: $\mathbf{p} \sim \mathbf{KP}$.

In practice, the absence of geometric distortions is rarely verified. Thus, to deal with the geometric aberrations inherent to the optical design of the lenses, many distortion models have been proposed. A commonly used representation is the Brown-Conrady's model (Duane, 1971) (called the Brown model in this paper) which maps the lens distortion via a polynomial function applied in the camera coordinate's system as follows:

$$\mathbf{p}_d = \begin{bmatrix} (k_1 r^2 + k_2 r^4 + k_5 r^6) x_u + (2k_3 x_u y_u + k_4 (r^2 + 2x_u^2)) \\ (k_1 r^2 + k_2 r^4 + k_5 r^6) y_u + (k_3 (r^2 + 2y_u^2) + 2k_4 x_u y_u) \end{bmatrix}, \quad (2)$$

where the distorted point at the location (x_d, y_d) – on the normalized camera coordinate system – is estimated from its undistorted counterpart $\mathbf{p}_u$ via the distortion parameters $\mathbf{k} = (k_1, k_2, k_3, k_4, k_5)$ and the distance to the distortion center expressed $r = \sqrt{x_u^2 + y_u^2}$. The parameters (k_1, k_2, k_5) map the radial distortion while (k_3, k_4) model the tangential distortion.

This model can accurately approximate a moderate amount of radial and tangential distortion. However, the Brown model remains relatively ineffective to represent wide field-of-view cameras which tend to exhibit larger radial distortions (Sturm and Ramalingam, 2011). To tackle this issue, the Kannala-Brandt model (Kannala and Brandt, 2006) proposes a different polynomial form:

$$\mathbf{p}_d = \begin{bmatrix} (\theta_d / r) x_u \\ (\theta_d / r) y_u \end{bmatrix}, \quad (3)$$

where $\theta_d = \theta(1 + k_1\theta^2 + k_2\theta^4 + k_3\theta^6 + k_4\theta^8)$ in which $\theta = atan(r)$.

In this work, we take advantage of the Brown model for perspective cameras and the Kannala model for the large field of view cameras (such as fisheye).

When a 3D point $\mathbf{P}$ is not expressed in the camera's referential, a prior rigid transformation of the point has to be applied. This rigid transformation is composed of a 3×1 translation vector $\mathbf{t} = [t_x, t_y, t_z]^T$ and a 3×3 orthogonal rotation matrix $\mathbf{R}$. For convenience, we also employ homogeneous transformation as a 4×4 matrix $\mathbf{M} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$. Thus, the projection of a 3D point $\mathbf{P}_w$ in the world referential can be expressed as follow:

$$\mathbf{p} = [x, y]^T = \mathcal{P}(\mathbf{R}\mathbf{P}_w + \mathbf{t}, \mathbf{K}, \mathbf{k}) = \mathcal{P}(\mathbf{M}\mathbf{P}_w, \mathbf{K}, \mathbf{k}), \quad (4)$$

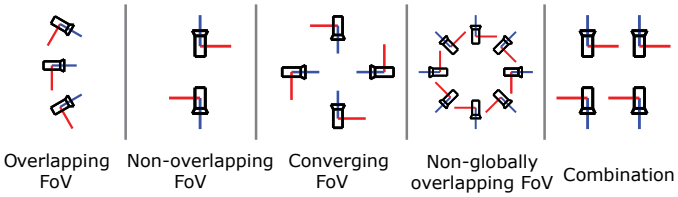


Fig. 3. Different multi-camera configurations.

where $\mathcal{P}$ is a 3D-to-2D projection operator mapping from homogeneous 3D coordinates to image coordinates given the intrinsic and extrinsic parameters. The rotation matrix can also be expressed in a compact vectorial manner such as quaternion or Rodrigues. In this work, we rely on the latter, expressing the rotation matrix $\mathbf{R}$ as its angle-axis representation $\mathbf{r}$.

2.2. Diverse multi-camera configurations

This subsection briefly describes different possible arrangements of multi-camera systems. In total, we can identify four main categories of multi-camera rigs (visualized in Fig. 3): the globally or non-globally overlapping field-of-view, the non-overlapping field-of-view, and the converging field-of-view configurations. A complex system can also be composed of a combination of different camera configurations. For instance, two individual groups of cameras can be used. While theoretical and practical tools have been proposed individually for the calibration of each configuration, no unified, generic, and robust approach has yet been introduced. In this paper, we develop an efficient and generic approach to calibrate any complex multi-camera systems.

2.3. Notations

An overview of our notations is presented in Fig. 4. The final goal of our calibration pipeline is to estimate the intrinsic and extrinsic parameters of N_c cameras given the observations of M_b checkerboards. The camera matrix and distortion parameters of the i^{th} camera are noted $\mathbf{K}_{c_i}$ and $\mathbf{k}_{c_i}$ respectively. Regarding the extrinsic parameters of the i^{th} camera, they are expressed with respect to a reference camera c_{ref} in the rig as $[\mathbf{R}^{c_i}_{c_{ref}} | \mathbf{t}^{c_i}_{c_{ref}}]$, or in homogeneous transformation: $\mathbf{M}^{c_i}_{c_{ref}}$. Similarly, the pose of the given i^{th} camera expressed in the referential of the j^{th} board observed at the t^{th} frame can be written ${}^t\mathbf{M}^{c_i}_{b_j}$. Multiple boards can be combined together as a single *object*, for instance in the example (Fig. 4) $o_0 = \{b_0, b_1\}$. Similarly to the boards, the k^{th} object can be expressed as o_k and its pose with respect to the camera observing it at the t^{th} frame can be written as ${}^t\mathbf{M}^{o_k}_{c_i}$. During our calibration process, we merge cameras sharing a common field-of-view in a single *camera group* expressed g. For instance, the group g_0 in Fig. 4 is formed by two cameras $g_0 = \{c_0, c_1\}$. The transformations between the groups of cameras without overlapping have to be estimated in the latest part of our process. The transformation between two camera groups g_0 and g_1 is denoted $\mathbf{M}^{g_1}_{g_0}$. Finally, the s^{th} 3D point on the j^{th} board is expressed $\mathbf{P}^s_{b_j}$ and its observation from the i^{th} camera at the t^{th} frame can be written ${}^t_{c_i}\mathbf{p}^s_{b_j}$.

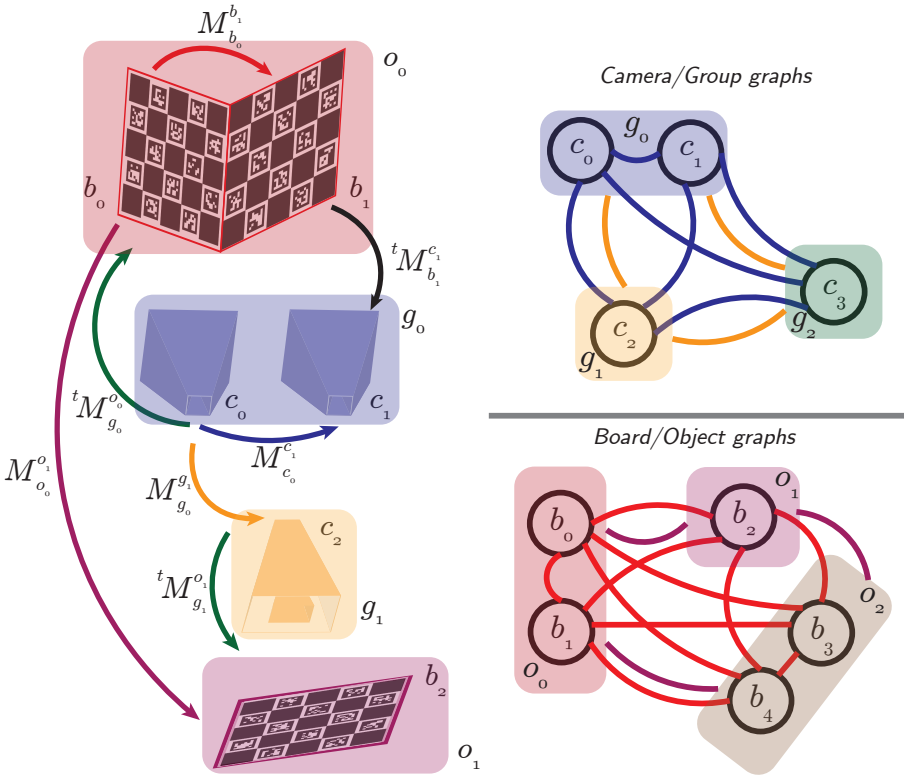


Fig. 4. Overview of a multi-camera system to be calibrated. In this figure, the inter-board, the inter-camera, inter-object, and inter-group transformations are depicted in red, blue, purple, and orange respectively. (a) A multi-camera system composed of three cameras $\{c_0, c_1, c_2\}$ observing three boards $\{b_0, b_1, b_2\}$ at a time t. Notice that the two cameras c_0 and c_1 have an overlapping field of view such that they form a camera group g_0 while the camera c_2 shares no FoV and forms a group g_1 alone. (b) The two graphs used in our pipeline. Note that the objects o_0 and o_1 as well as the camera groups g_0 and g_1 are similar to the configuration of the left figure. A third object and camera group has been included to show the expandability of the approach.

	Perspective	Fisheye	Hybrid	Stereo	Multi-Camera	Non-Overlapping	Converging FoV	No boards
Scaramuzza et al. (2006)	✓	✓						
Caron and Eynard (2011)	✓	✓	✓	✓				
Kalibr Rehder et al. (2016)	✓	✓	✓	✓	✓			
Mei and Rives (2007)	✓	✓						
Bouguet (2004)	✓			✓				
Itseez (2015)	✓	✓		✓				
Lin et al. (2020)	✓	✓		✓	✓	✓		✓
Li et al. (2013)	✓	✓	✓	✓	✓			
Heng et al. (2013)	✓	✓	✓	✓	✓	✓		✓
Liu et al. (2016)	✓			✓	✓	✓	✓	
Ours	✓	✓	✓	✓	✓	✓	✓	

Table 1. Summary of existing camera calibration toolboxes.

3. Bibliography

Camera calibration is the initial stage of most 3D reconstruction techniques, thus, it has been an important research topic since the very beginning of photogrammetry. One of the first practical camera calibration pipelines has been proposed by Tsai (1987). This approach requires a single image of the calibration pattern assuming its coplanarity with the camera plane. While this assumption is difficult to ensure in practice, the method proposed by Zhang (2000) only needs multiple observations of a checkerboard without any restriction regarding its position. The Zhang's calibration pipeline is currently the most commonly used strategy to estimate the intrinsic parameters of a perspective camera. For instance, it is implemented in the Bouguet's toolbox (Bouguet, 2004), OpenCV (Itseez, 2015), and Matlab. To deal with cameras with large radial distortions, multiple ad hoc solutions have also been proposed (Scaramuzza et al., 2006; Mei and Rives, 2007).

The extension of single-camera calibration techniques to a stereo-vision system with large overlapping field-of-view is trivial and has been implemented in most camera calibration toolboxes (Bouguet, 2004; Mei and Rives, 2007). However, these toolboxes do not extend to the calibration of more than two cameras. This limitation can be partly explained by the type of checkerboard utilized. Indeed, to calibrate a multi-camera system, indexed observations of the 3D points on the board should be visualized simultaneously by different cameras. Using a traditional checkerboard, the entire board needs to be visible (to estimate the indexing of the points) which is hardly applicable when a large number of cameras is utilized and/or if the baseline between the cameras is large. To cope with this limitation, Li et al. (2013) propose to use a randomly textured calibration pattern on which unique keypoints can be detected to perform the calibration of a vision system even when a limited overlapping between fields of views is available.

More recently, the development of effective fiducial marker systems allows estimating the index of the observed cor-

ners without the need to visualize the entire board. Among well-known Augmented Reality (AR) markers, we can mention Charuco (Itseez, 2015) and AprilTag (Olson, 2011; Wang and Olson, 2016) which have been widely used for multi-camera calibration systems. Such specific calibration markers have drastically eased the calibration of complex vision systems (Xing et al., 2017; Rehder et al., 2016; Strauß et al., 2014).

The previously mentioned calibration approaches assume that at least a partially overlapping field of view between the cameras is available (Rehder et al., 2016) or that the checkerboards can be observed together such that they can be merged in a single 3D calibration object (Strauß et al., 2014). However, they do not allow generic non-overlapping and converging systems calibration without specific a-priori. To conduct the calibration of non-overlapping cameras, many approaches following the hand-eye estimation strategy have been proposed (Tsai et al., 1989). In the literature, this type of calibration is often achieved under certain assumptions (Lébraly13 et al., 2010; Im et al., 2016): known intrinsic parameters; one board per camera is used (and this board remains the same for the entire sequence); the motion used for calibration should not be degenerate (translation in each direction).

More complex calibration strategies, involving additional hardware, have also been developed for the calibration of non-overlapping systems. For instance, in (Kumar et al., 2008), a mirror is utilized to virtually obtain a shared view of the calibration board. A more flexible technique has also been proposed by Zhao et al. (Zhao et al., 2018) where an external camera is used to compute the displacement of the multi-camera rig to be calibrated. Despite their complexity and lack of scalability, these approaches have the advantage to be more robust against degenerate motions than their hand-eye-based counterparts.

However, hand-eye-based approaches tend to be more generic and do not require any specific and cumbersome setups. A good illustration of the versatility of hand-eye-based approaches is the toolbox "Caliber" (Liu et al., 2016) which shares similarities with our technique. This toolbox has also been designed to be theoretically compatible with any configuration of cameras. However, in practice, this technique suffers many technical shortcomings. For instance, it is not compatible with fisheye or hybrid (combination of perspective and fisheye cameras) vision systems. Moreover, no fiducial markers are used which implies that many human manipulations of the data are required. On the contrary, our proposed toolbox can deal with various camera distortion models and is fully automatic. Furthermore, neither the motion of the camera rig nor the number of boards is limited by our strategy.

In this work, we also rely on a hand-eye calibration strategy to estimate the pose between the non-overlapping camera groups. However, we have extended it with multiple strategies to improve the stability and accuracy of the proposed approach. Another notable difference with (Liu et al., 2016) is the overall structure of our method. In our case, we design a hierarchical strategy where problems are solved gradually with a systematic non-linear refinement leading to a good convergence. Finally, we focus our method to be particularly robust by including a RANSAC process (prevents wrong markers' detection) and robust non-linear optimizations – allowing to deal with outliers.

On the application side, the calibration of non-overlapping cameras is particularly critical for automotive vision systems to provide an all-around view of the scene. For the sake of practicability, numerous strategies have been proposed to simplify the calibration of such systems without the need for calibration boards. A seminal work has been proposed in (Heng et al., 2013), where the displacement of each camera is estimated (using visual odometry) to calibrate the system via a hand-eye calibration technique. While this approach does not require setting up multiple calibration boards, additional information to estimate the scale of the displacement (i.e. wheel encoder or stereo vision systems) is needed. Moreover, the accuracy obtained with such a strategy is scene-dependent and is relatively complex to deploy due to a large number of stages involved. To simplify this calibration process, Ataer-Cansizoglu et al. (2014) take advantage of the prior reconstruction of an arbitrary *calibration scene* (from a single RGB-D camera) to calibrate a set of non-overlapping cameras. This approach - simple and effective - provides 3D metric scale calibration estimation, but, it requires a prior reconstruction of the scene and cannot guarantee repeatable calibration results.

The previously described techniques assume pre-calibrated intrinsic camera parameters. To ease the automatic calibration process further, Lin et al. (2020) propose to use a radial projection model to estimate the poses of the cameras in a pre-reconstructed 3D scene. The advantage of this strategy is that no prior intrinsic parameter is needed to estimate the extrinsic and intrinsic parameters of the cameras. While this approach is practical and versatile, a scene reconstruction at a metric scale remains a complex task that can be affected by drifts and artifacts. Moreover, the approach developed by Lin et al. (2020) suffers from structural limitations related to the pose estimation via the radial projection model. For instance, the vision system cannot be calibrated unless at least two cameras have a non-parallel principal axis. Admittedly, the checkerboard-free strategies are very desirable for practical tasks but remain complex to deploy in practice due to their lack of accuracy, use case limitation (i.e. cannot be utilized for converging field-of-view camera calibration), and repeatability issues.

While this literature covers the problem of multiple camera systems used in robotics, another type of multi-camera system that we call converging camera system is often needed for single-shot 3D scanner (Pesce et al., 2015) (see Fig. 3). This kind of camera system can hardly be calibrated using traditional planar patterns and often requires 3D calibration patterns. Only a few approaches address this particular problem. A representative work is (Forbes et al., 2002) where the geometry of a 3D cube is refined using multiple observations to estimate the relative poses between the cameras composing the system. The major limitation of this work is the need for prior information regarding the position of the 3D points on the calibration object. In our work, this object structure is automatically estimated without any prior 3D information provided by the user. Worth noting that such vision system can also be calibrated with hand-eye calibration techniques. However, such strategies require each camera to observe a unique board during the calibra-

tion process which drastically restricts the variety of possible motions, leading to biased calibration results. On the contrary, our technique can take advantage of the entire 3D object, allowing a wider range of motions. Therefore, our system simplifies the calibration process and avoids degenerated configurations.

Our solution is a fully functional calibration toolbox called "MC-Calib". To underline the relevance of this software, we provide a quick overview of the existing multi-camera calibration toolboxes with their inherent limitations in Table 3.

4. Methodology

In this section, we describe the technical details of the proposed multi-camera calibration pipeline. Before digging into the detailed description of each stage composing our strategy, we propose an overview of the entire calibration framework. First of all, a Charuco board detection (Sec. 4.1) is performed for all the images acquired by the cameras and the detected 2D location are stored. These 2D observations and their respective 3D locations (expressed in their board referential) are used to initialize the intrinsic parameters (Sec. 4.2) for every camera. After estimating the internal parameters of the cameras, the pose of each camera with respect to the observed boards is estimated (Sec. 4.3) via a perspective-n-point technique.

The inter-board transformations between all boards visible in a single image are computed (Sec. 4.4) to merge the boards sharing a co-visibility into 3D objects (a 3D object is a set of 3D boards). After the refinement of the 3D object structures, we group the cameras (Sec. 4.5) which have seen similar 3D objects synchronously via a graph-based strategy. At this stage, if all the cameras in the system share a globally or non-globally overlapping field of view, the calibration is finalized via a final bundle adjustment (Sec. 4.7). If multiple camera groups remain, a non-overlapping camera group calibration (Sec. 4.6) is performed between each pair of groups. This estimation is used to merge all the camera groups and the 3D objects before being refined to obtain the entire parameters of the camera system.

4.1. Checkerboard detection and keypoints extraction

The initial stage of our calibration process is the detection of the checkerboards in the images and the precise localization of their 2D corners (see Fig. 1). To deal with any complex setups, we propose to utilize fiducial checkerboard markers. Specifically, we take advantage of the CharucoBoard (Itseez, 2015) mixing a standard planar checkerboard pattern with ArUco fiducial markers (Garrido-Jurado et al., 2014).

During this stage, all the images from all the cameras are processed to store their 2D keypoints location and corresponding 3D points in their board's referential. This detection is critical since the entire calibration of the system strongly depends on the robustness and accuracy of these 2D keypoints. Thus, to improve the accuracy of the calibration, we apply an effective corner refinement process Ha et al. (2017). To avoid degenerated configuration, we additionally apply a collinearity check. Moreover, to improve the overall robustness, the boards with less than a certain percentage of visible keypoints are discarded from further consideration – we typically set this threshold to 40%. Note that this threshold can be adjusted by the user in case of small overlapping FoV camera systems calibration.

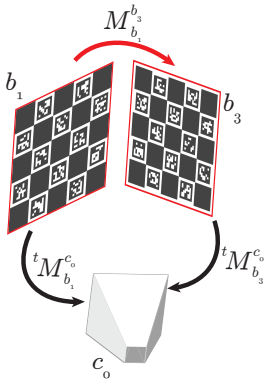


Fig. 5. Board pose estimation at a time t. Here, a camera c_0 can see two boards b_1 and b_3 in a single frame and estimate their inter-board pose.

4.2. Intrinsic parameters initialization

For each i^{th} camera c_i, we collect the 3D↔2D correspondence pairs from all the images containing checkerboards. These matches are used to initialize the intrinsic parameters $\mathbf{K}_{c_i}$ and distortion coefficient $\mathbf{k}_{c_i}$. For perspective cameras, we adopt the well-known (Zhang, 2000) calibration technique (Brown distortion model) while the calibration of fisheye cameras is accomplished with the implementation available in OpenCV (Itseez, 2015) (Kannalla distortion model). This initialization is relatively slow if a large number of images are used, therefore, we subsample the number of images by randomly selecting a subset of 50 boards observations per camera. If less than 50 board observations are available, all images are utilized. Notice that the intrinsic parameters are refined using all the images in the next stage.

For certain complex scenarios involving a large number of non-overlapping cameras, it is sometimes tedious to acquire enough diversified viewpoints to reach an accurate intrinsic calibration of the individual cameras. Therefore, our toolbox can also use pre-computed intrinsic parameters provided by the user. As another functionality, it is compatible with hybrid systems mixing fisheye and perspective cameras under the condition the user specifies the type of each camera in the system.

4.3. Board pose estimation and intrinsic refinement

Given the initial intrinsic parameters — computed from the previous stage 4.2, we estimate the relative pose of all the cameras for each observed board. An illustration of this process for an arbitrary frame t is visible in Fig. 5. Notice that a single image can contain multiple boards, for instance the camera c_0 at frame t sees two boards b_1 and b_3 simultaneously, thus both transformation ${}^t\mathbf{M}^{c_0}_{b_1}$ and ${}^t\mathbf{M}^{c_0}_{b_3}$ are computed and stored. The estimation of these poses is achieved with a PnP algorithm (Gao et al., 2003) wrapped in a RANSAC robust estimation process (Fischler and Bolles, 1981). This RANSAC stage is intended to remove very large outliers only (e.g. error superior to 10 pixels reprojection error), improving the overall robustness of our pipeline. The inlier points are then used to

refine the pose estimation of the camera w.r.t. to the board via a Levenberg-Marquardt non-linear refinement. Considering a camera c_i at the frame t, its pose w.r.t. the board b_j is refined by minimizing the following reprojection error function:

$$\min_{{}^{t}\mathbf{r}_{b_j}^{c_i},\,{}^{t}\mathbf{t}_{b_j}^{c_i}} \sum_{s=1}^{S} \left\| {}^{t}_{c_i}\mathbf{p}^{s}_{b_j} - \mathcal{P}({}^{t}\mathbf{M}_{b_j}^{c_i}\mathbf{P}^{s}_{b_j}, \mathbf{K}_{c_i}, \mathbf{k}_{c_i}) \right\|^2, \tag{5}$$

where S is the number of corner visible on the board. Given these initial guesses for the extrinsic and intrinsic parameters, we refine them together for all the images acquired by each camera individually via the following cost function:

$$\min_{{}^{t}\mathbf{r}_{b_j}^{c_i},\,{}^{t}\mathbf{t}_{b_j}^{c_i},\mathbf{K}_{c_i},\mathbf{k}_{c_i}} \sum_{t=1}^{T}\sum_{j=1}^{M_b}\sum_{s=1}^{S} \left\| {}^{t}_{c_i}\mathbf{p}^{s}_{b_j} - \mathcal{P}({}^{t}\mathbf{M}_{b_j}^{c_i}\mathbf{P}^{s}_{b_j}, \mathbf{K}_{c_i}, \mathbf{k}_{c_i}) \right\|^{H} \quad \forall i, \tag{6}$$

where T and M_b are respectively the number of frames and the number of boards observed by the i^{th} camera. $\|\cdot\|^H$ is the Huber loss function to ensure robustness against outliers.

4.4. Boards grouping into objects

In the previous stage, we estimate the relative pose of all the cameras for every single board observed. We now attempt to find the relative pose between the boards to gather them into 3D objects. A 3D object is formed of multiple planar calibration boards, for instance in the illustration Fig. 4, the object o_0 is formed by two planar targets b_0 and b_1. If two or more boards are visible in a single image their pair-wise relative poses can be estimated. For instance, in Fig. 5 the relative pose between the two boards ${}^{t}\mathbf{M}_{b_1}^{b_3}$ can be computed by the matrix multiplication ${}^{t}\mathbf{M}_{b_1}^{b_3} = ({}^{t}\mathbf{M}_{b_3}^{c_0})^{-1}\,{}^{t}\mathbf{M}_{b_1}^{c_0}$.

For the sake of robustness, we gather the measurements from all the images where pairs of boards are visible together in the same image and compute their average inter-board rotation and translation. For instance, if the boards b_0 and b_1 have been observed together over 10 images – by one or more cameras in the rig – then the inter-board relationship will be the robust average of these 10 measurements. This strategy allows to gain significant robustness and can allow the system to be calibrated despite potential outliers.

This averaging provides a solid prior estimation of the inter-board poses. To construct the 3D objects from these inter-board poses, they are stored in a directed weighted graph as shown in Fig. 4. Connected components of this graph constitute the 3D objects. For each object, a reference board is empirically selected as the board with the lowest index. For instance, in Fig. 4 the reference board of o_0 and o_2 are b_0 and b_3 respectively. Thus, the boards poses among their respective object is expressed as $\mathbf{M}_{b_{ref}}^{b_j}$ for the j^{th} board in its object. When the 3D object is constituted by more than two boards, a Dijkstra shortest path algorithm (Dijkstra, 1959) is used to determine the best transformation composition (to express each board in the object reference board coordinate system) assuming the edges of the graph contain the inverse of the number of observations ($1/Nb_{observation}$) where both boards have been seen together. This strategy welcomes robust paths with many observations.

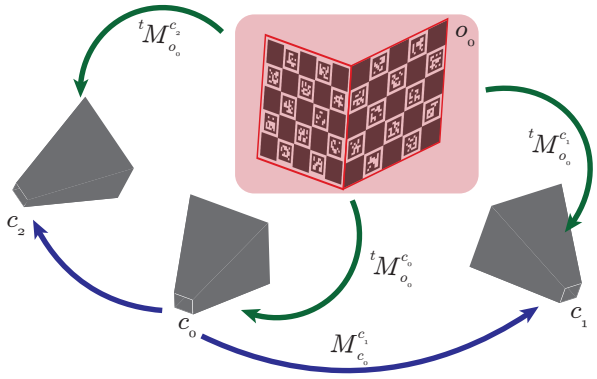


Fig. 6. Illustration of multiple cameras observing a single 3D object at time frame t. Since these 3 cameras share an overlapping field of view they can be clustered together to become a single camera group (all the 3 grey cameras are automatically merged into the same group by our strategy).

Finally, the 3D structure of each 3D object is refined in a non-linear fashion by minimizing the following reprojection error:

$$\min_{\mathbf{r}_{b_{ref}}^{b_j},\,\mathbf{t}_{b_{ref}}^{b_j}} \sum_{i=1}^{N_c}\sum_{j=1}^{M_b}\sum_{t=1}^{T}\sum_{s=1}^{S} \left\| {}^{t}_{c_i}\mathbf{p}^{s}_{b_j} - \mathcal{P}({}^{t}\mathbf{M}_{b_{ref}}^{c_i}\mathbf{M}_{b_j}^{b_{ref}}\mathbf{P}^{s}_{b_j}, \mathbf{K}_{c_i}, \mathbf{k}_{c_i}) \right\|^{H}. \tag{7}$$

4.5. Grouping cameras as camera groups

After the creation of the 3D objects resulting from the merging of the boards, the pose of the cameras w.r.t. the objects for all frames is estimated with a PnP algorithm similarly to Sec. 4.3. For instance, in Fig. 6 the pose of the camera c_2 w.r.t. the object o_0 at a frame t is expressed as ${}^{t}\mathbf{M}_{o0}^{c_2}$. Since these three cameras can see a single object simultaneously, they can be merged into a single camera group $g_0 = \{c_0, c_1, c_2\}$. Following the strategy explained in Sec. 4.4, the inter-camera pose estimations for multiple frames are averaged and the cameras observing an object simultaneously are grouped as camera groups (see Fig. 6). This grouping is performed via the camera graph, depicted in Fig. 4, where each connected component forms a camera group. Once again, we wish to express the camera pose in the referential of the reference camera of the group (camera with the lowest index value). To initialize the cameras' poses w.r.t. this reference camera, the (Dijkstra, 1959) algorithm is used to determine the path (maximizing the number of observations) in the graph. Finally, all the camera groups are refined individually using a Levenberg-Marquardt method by minimizing the following cost function:

$$\min_{\mathbf{r}_{c_{ref}}^{c_i},\,\mathbf{t}_{c_{ref}}^{c_i}} \sum_{i=1}^{N_g}\sum_{k=1}^{M_o}\sum_{t=1}^{T}\sum_{s=1}^{S_o} \left\| {}^{t}_{c_i}\mathbf{p}^{s}_{b_j} - \mathcal{P}(\mathbf{M}_{c_{ref}}^{c_i}\,{}^{t}\mathbf{M}_{o_k}^{c_{ref}}\mathbf{P}^{s}_{o_k}, \mathbf{K}_{c_i}, \mathbf{k}_{c_i}) \right\|^{H}, \tag{8}$$

where N_g is the number of cameras in the camera group, while M_o and S_o are the number of objects and the number of points in the object respectively.

4.6. Non-overlapping camera groups estimation

If a single camera group remains, the calibration can be finalized as described in sec. 4.7. Otherwise, the existence of multiple camera groups implies that they do not share a common field

of view. For each pair of remaining camera groups, their inter-pose is estimated via a hand-eye calibration approach (Tsai et al., 1989). Thus, these pairs of camera groups can be merged into a single final camera group. In the remainder of this section, we describe this process in detail, including, the hand-eye pose estimation from a pair of non-overlapped camera groups and our robust bootstrapped initialization technique.

4.6.1. Hand-eye calibration for non-overlapping cameras

Considering two camera groups g_0 and g_1 (as depicted in Fig. 7), their inter-pose $\mathbf{M}_{g_0}^{g_1}$ can be calculated using a hand-eye calibration strategy. Assuming each group can visualize one object across multiple frames, the camera groups' displacements can be estimated individually. For instance, if the group g_0 and g_1 capture an object o_0 and o_1 respectively across two frames t_0, t_1, then their displacements can be estimated as follow:

$$ {}^{t_1}_{t_0}\mathbf{M}_{g_0} = {}^{t_1}\mathbf{M}_{o_0}^{g_0\,t_0}\mathbf{M}_{g_0}^{o_0} \tag{9} $$

$$ {}^{t_1}_{t_0}\mathbf{M}_{g_1} = {}^{t_1}\mathbf{M}_{o_0}^{g_1\,t_0}\mathbf{M}_{g_1}^{o_0}. \tag{10} $$

As a result, the relationship linking camera groups can be written ${}^{t_1}_{t_0}\mathbf{M}_{g_1}\mathbf{M}_{g_0}^{g_1} = \mathbf{M}_{g_0\,t_0}^{g_1\,t_1}\mathbf{M}_{g_0}$, without loss of generality, this relation can be generalized for all the frames:

$$ \forall t_i \in [1\cdots T], \forall t_j \in [1\cdots T],\ {}^{t_j}_{t_i}\mathbf{M}_{g_1}\mathbf{M}_{g_0}^{g_1} = \mathbf{M}_{g_0\ t_i}^{g_1\ t_j}\mathbf{M}_{g_0}. \tag{11} $$

This problem takes the form of a system $\mathbf{AX} = \mathbf{XB}$ which can be resolved using a hand-eye calibration technique. In this work, we employ the approach proposed by Tsai et al. (1989) which consists in a hierarchical resolution of the problem: 1) rotation estimation first and 2) translation computation. To perform the rotation estimation, the authors propose to utilize a variation of the Rodrigues angle-axis representation such that the rotation vector $\mathbf{r}_{g_0}^{g_1}$ can be linearly resolved as follows:

$$ \left[{}^{t_j}_{t_i}\mathbf{r}_{g_1} + {}^{t_j}_{t_i}\mathbf{r}_{g_0}\right]_\times \mathbf{r}_{g_0}^{g_1\prime} = {}^{t_j}_{t_i}\mathbf{r}_{g_0} - {}^{t_j}_{t_i}\mathbf{r}_{g_1}, \tag{12} $$

where the operator $[\cdot]_\times$ stands for the transformation of a 3D vector to a skew-symmetric matrix and $\mathbf{r}_{g_0}^{g_1} = (2\mathbf{r}_{g_0}^{g_1\prime})/(1+|\mathbf{r}_{g_0}^{g_1\prime}|^2)$. After the rotation being solved, the translation $\mathbf{t}_{g_0}^{g_1}$ can be computed via the following set of linear equations:

$$ \left({}^{t_j}_{t_i}\mathbf{R}_{g_1} - \mathbf{I}\right)\mathbf{t}_{g_0}^{g_1} = \mathbf{R}_{g_0\ t_i}^{g_1\ t_j}\mathbf{t}_{g_0} - {}^{t_j}_{t_i}\mathbf{R}_{g_1}, \tag{13} $$

where $\mathbf{I}$ is the identity matrix. Note that at least two pairs of motion are needed to perform this hand-eye calibration. For further details, we invite the reader to refer to Tsai et al. (1989).

Note that, this initial calibration is achieved using only the two objects that have been seen the largest number of times simultaneously by both camera groups.

4.6.2. Best view selection and bootstrapped initialization

The hand-eye calibration strategy can be applied for all possible combinations of frames but the complexity of the problem is growing quadratically with the number of frames which is problematic (computational time-wise) for large video sequences. Moreover, successive frames exhibit similar poses which does not contribute much to the final solution. Furthermore, using all the frames at once can be problematic if the set of frames contains outliers. Therefore, we propose an effective manner to select the best views to calibrate each pair of non-overlapping camera groups in the system in a fast and robust manner.

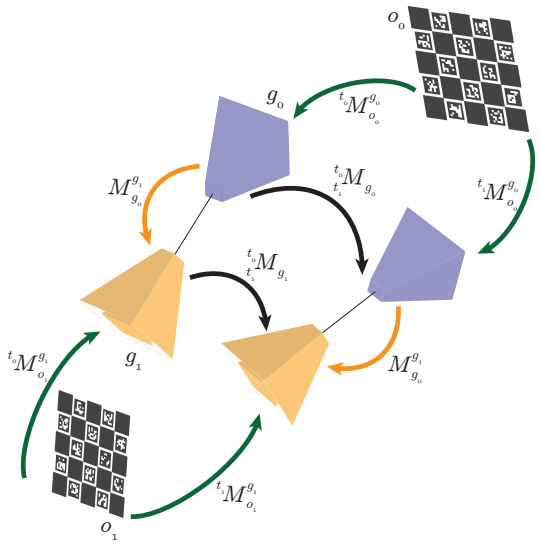


Fig. 7. Representation of two camera groups (g_0 in blue and g_1 in yellow) with non-overlapping field-of-view observing one object each (for the sake of clarity and simplicity, each camera group contains a single camera and each group is composed of a single board in this example).

Our best view selection is designed to maximize the diversities of views used for the calibration to avoid degenerated configurations and to improve the robustness against outliers. For this purpose, for each frame, we concatenate the translational component of both camera groups to cluster the frames as depicted in Fig. 8. Therefore, the most similar poses are assigned to the same cluster. In this situation, the rotational component does not need to be considered to ensure the diversity of the poses across clusters since a rotation of one of the groups will inevitably lead to a translation of the second group. In our framework, a k-means clustering (Lloyd, 1982) technique is utilized with the number of clusters fixed to 20.

After this initial clustering, we initiate our bootstrapping strategy which consists in the successive estimation of the inter-group pose via the selection of multiple mini-batches of frames. Specifically, for each iteration of our bootstrapping algorithm, 6 clusters are randomly sampled (from the initial 20 clusters), among each of these 6 clusters one pose is chosen randomly. The resulting set of 6 pair of poses is used to perform the hand-eye calibration (as described in Sec. 4.6) such that the inter-group pose can be computed. The validity of the set is then evaluated by estimating the consistency of the rotational solution provided by the hand-eye calibration algorithm. If the maximum rotational error in the set is superior to 5°, the solution is rejected and, if the set is consistent, this result is stored. This mini-batch hand-eye estimation is repeated 200 times leading to a set of plausible inter-group poses. Finally, the estimation of the inter-group pose is obtained by computing the median value of each translation and rotation (Rodrigues representation) element which passed the rotational test. This initial solution is, thereafter, refined in a non-linear manner. Our bootstrapped initialization procedure has proved to be very effective against

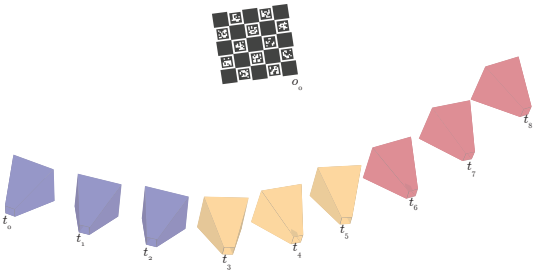


Fig. 8. Example of the resulting clustering (3 clusters: blue, yellow, and red) for a single camera orbiting around a calibration board. We can see that nearby frames belong to the same cluster. In practice, a concatenation of two non-overlapping cameras or camera groups is used for this clustering.

outliers and noisy measurements.

4.7. Merging camera groups and Bundle adjustment

After the initial poses between all the non-overlapping pair of camera groups are estimated, the camera groups and the objects are merged using a similar graph strategy described in Section 4.4. Finally, the entire system (relative position between all boards, camera poses, and the intrinsic parameters) can be refined to minimize the reprojection error in all frames:

$$\min_{\mathbf{r}^{b_j}_{b_{ref}},\mathbf{t}^{b_j}_{b_{ref}},\mathbf{r}^{b_i}_{c_{ref}},\mathbf{t}^{b_i}_{c_{ref}},\mathbf{K}_{c_i},\mathbf{k}_{c_i}} \sum_{i=1}^{N_c}\sum_{j=1}^{M_b}\sum_{t=1}^{T}\sum_{s=1}^{S} \Big\| {}^{t}_{c_i}\mathbf{p}^{s}_{b_j} - \mathcal{P}(\mathbf{M}^{c_{ref}}_{c_i}\,{}^{t}\mathbf{M}^{c_i}_{b_{ref}}\mathbf{M}^{b_{ref}}_{b_j}\mathbf{P}^{s}_{b_j}, \mathbf{K}_{c_i}, \mathbf{k}_{c_i}) \Big\|^{H}. \tag{14}$$

Our hierarchical calibration strategy provides initialization that is in the vicinity of convergence leading to very stable and accurate results.

5. Experiments

This section contains a large number of assessments on real and synthetic data. Various use cases are proposed to reflect a broad spectrum of scenarios commonly faced in practice. Multiple metrics are used to evaluate the quality of the retrieved parameters. The rotational error is calculated as follows:

$$\epsilon_R = acos(\frac{1}{2}(tr(\mathbf{R}^T_{est}\mathbf{R}_{GT}) - 1)), \tag{15}$$

where $\mathbf{R}_{est}$ is the estimated rotation matrix and $\mathbf{R}_{GT}$ the ground truth rotation. Regarding the translational and the internal parameters' (principal point pp and focal length) errors, it is the euclidean distance between the ground truth and the estimated values. The reported reprojection error is the mean of the Euclidean distance between the detected and reprojected points for all the corners observed by all cameras. Note that for the sake of conciseness we do not provide comparative results for the aspect ratio λ while the skew factor is assumed to be zero.

Table 2. Average intrinsic and extrinsic errors over all the cameras on synthetic data generated by image rendering.

Error \ Seq	Seq01	Seq02	Seq03	Seq04	Seq05
focal (px)	27.601	2.229	27.611	27.648	0.124
pp (px)	0.396	2.060	0.514	0.464	0.718
Rotation (°)	0.002	0.056	0.002	0.005	0.046
Translation (m)	0.000	0.006	0.000	0.000	0.002
Reprojection (px)	0.022	0.023	0.014	0.017	0.090

5.1. Experiments on rendered images

To provide a quantitative estimation regarding the accuracy and versatility of the proposed technique, we use Blender (Community, 2018) to generate a synthetic dataset composed of 5 different calibration scenarios (see Fig. 9): 1) Stereo system (2 cameras); 2) Non-globally overlapping vision system (3 cameras); 3) Non-overlapping system (4 cameras); 4) Unbalanced non-overlapping vision system (3 overlapping cameras and one non-overlapping camera); 5) Converging vision system (4 cameras). The dataset, the codes, and Blender 3D models used for its creation are available to the public via the following link[1]. For each sequence, 100 synchronized and distortionless frames per camera have been captured at a resolution of 1824×1376 px. A set of representative synthetically generated images is available in Fig. 9. For this experiment, the field of view of the synthetic cameras is fixed at 65° of horizontal FoV. The mean calibration error against the ground-truth (over all the cameras in the rigs) are available in Table 2.

Seq01: Stereo. While the proposed technique is designed to calibrate complex vision systems, it can be employed for the calibration of rather simple and common vision rig such as monocular or stereo vision systems. Such calibration can be achieved with our toolbox using a single calibration board. To challenge our technique, we utilize 3 individual boards. In this scenario, our toolbox outputs highly accurate results with a submillimetric translational error and a very low reprojection error.

Seq02: Non-globally overlapping camera system. For this experiment, we simulate an omnidirectional vision system that shares similarities with (Schroers et al., 2018). This system is composed of 5 cameras arranged in a semi-circle (see Fig. 9(b)). A single calibration board is used for the entire calibration of the vision rig, since each camera shares a partial FoV with its neighbors, the relative poses of the calibration can be achieved by chaining all the transformations. This process is automatically achieved in our calibration pipeline. The difficulty of such calibration scenario is the possibility to accumulate a drift between the reference camera and the other cameras in the system. Despite this challenging scenario, our calibration technique is able to reliably estimate the camera poses in the system. We can notice a higher mean translational error in this sequence which mostly comes from one camera located on the extreme left of the system with partial visibility of the boards.

[1]link to rendered images dataset: `https://bosch.frameau.xyz/index.php/s/pLc2T9bApbeLmSz`

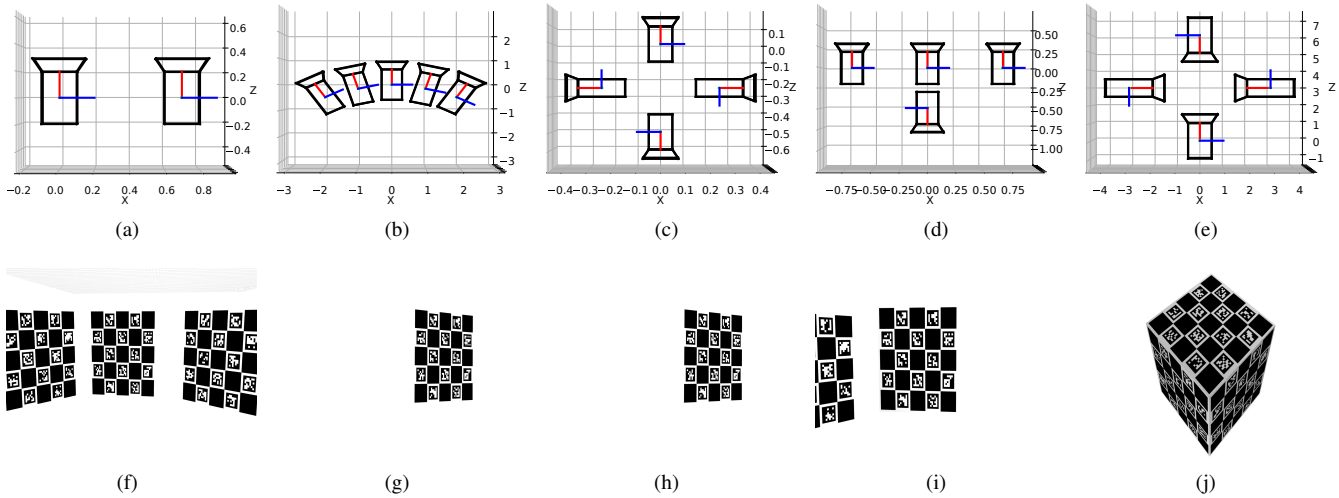


Fig. 9. Calibration results from synthetically rendered images: (a) Seq01: Stereo, (b) Seq02: Non-globally overlapping, (c) Seq03: Non-overlapping, (d) Seq04: Unbalanced, (e) Seq05: Converging vision system. (f-j) Sample of rendered images from each sequence.

Seq03: Non-overlapping camera system. Our calibration toolbox can be used to calibrate any multi-camera vision system. In particular, we propose a robust strategy for the calibration of non-overlapping camera systems (see Fig. 9(c)). In this experiment, we proposed one of the most common multi-camera systems used to obtain an all-around view (as depicted in Fig. 9(c)). This multi-camera vision system is surrounded by a set of eight calibration boards placed in a circular manner such that the boards can be visible by multiple cameras simultaneously. Despite the limited amplitude of motions used for this calibration, the obtained results are very close to the ground truth with nearly no translational or rotational error (see Table 2).

Seq04: Unbalanced non-overlapping vision system. To evaluate the applicability of our calibration strategy on an unbalanced non-overlapping vision system, we simulate 3 overlapping cameras on one side and a single camera pointing in the opposite direction (see Fig. 9(d)). This scenario is complex since a wrongly initialized calibration may lead to a wrong convergence of the calibration due to the unbalanced reprojection error on both sides of the system. Our technique is both robust and effective even for such a specific scenario. This satisfying performance can be explained by the design of our method. Specifically, our method is built to solve the calibration problem in a progressive step-by-step manner where each step is designed to provide an accurate initialization to the next one.

Seq05: Converging vision system. Most existing calibration toolboxes are incompatible with converging vision systems. Our toolbox can deal with such scenarios by estimating the structure of the 3D calibration object. While our method can function with any calibration object composed of multiple planar boards, in this experiment we propose to simulate the most common 3D calibration object: a cube composed of 6 planar boards. A group of 4 converging cameras are orbiting randomly around the cube such that every faces of the cube can be observed. The results presented in Table 2 demonstrates sub-pixel reprojection error and highly accurate camera pose estimation.

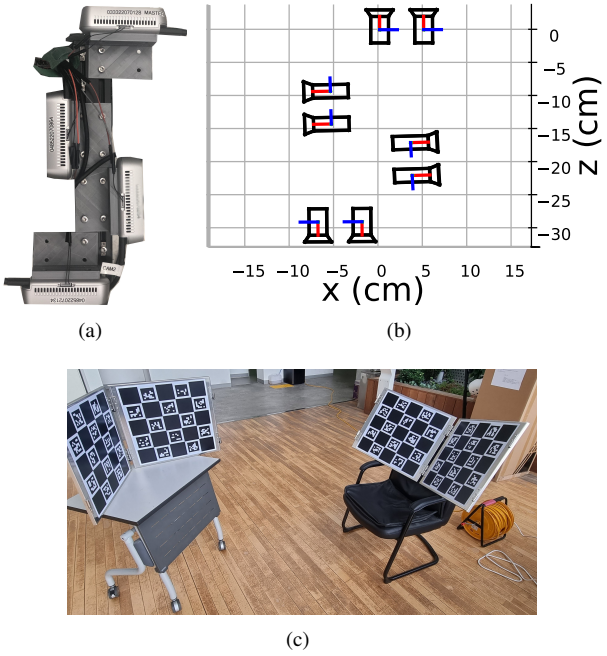
5.2. Real vision system experiments

To demonstrate the effectiveness of the calibration approach under realistic scenarios, we propose to calibrate diverse vision systems ranging from stereo to multiple groups of non-overlapping cameras. Moreover, the proposed scenarios involve a different number of calibration boards and 3D objects. All the sequences used in this paper can be downloaded freely[2].

The stereo vision system calibrated in this experiment is a ZED camera capturing synchronized pair of 1280×720px resolution images. For the multi-camera system configurations, we use up to 4 synchronized Intel Realsense D415i RGB-D cameras. Specifically, we utilize 2 infrared sensors of each camera (spatial resolution of 1280×720px). While the RGB sensor can be used together for the calibration, the large motion blur and the rolling shutter of this color camera disqualified it for this experiment. Due to hardware limitations, the miss-synchronization of RGB-D camera can reach up to 5ms. This delay does not seem to cause a significant problem during the calibration process since a low reprojection error has been reported. Finally, the hybrid stereo-vision system is composed of two PointGrey Flea3 cameras FL3-U3-13E4C-C (1280×512px spatial resolution). The left one is equipped with a lens (LM3NCM) providing a 90° horizontal field of view with low radial distortion. On the right camera, we install a fisheye lens (Fujinon FE185C046HA-1) yielding 182° horizontal field of view and very large radial distortion. These two cameras are spaced by a baseline of 20cm. All the experiments presented in this paper have been conducted on a desktop computer equipped with 32GB of RAM and a CPU i7-6800K.

Stereo vision system. To validate our method, we calibrate a stereo camera to allow comparison against a widely used strategy proposed in (Bouguet, 2004). The stereo ZED camera is calibrated with a single board in both cases. While we utilize

[2]link to real images dataset: `https://bosch.frameau.xyz/index.php/s/fqtFij4PNc9mp2a`



(a) (b)

(c)

Fig. 10. Experimental setup for our non-overlapping vision system. (a) Camera rig, (b) obtained calibration result, (c) boards used for calibration.

a video sequence consisting of 900 frames per camera for our methods only 50 frames are employed for the calibration using Bouguet (2004) since the detection of the board is manual. Our calibration leads to a mean reprojection error of 0.39 pixels while (Bouguet, 2004) suffers from a mean error of 0.44 pixels. This metric by itself is not sufficient to be conclusive regarding the accuracy of the method, therefore, we provide a comparison on the estimated parameters in Table 3. Both toolboxes result in similar parameters with a translational difference of 0.5mm and an insignificant rotational difference. This experiment suggests the ability of our strategy to calibrate stereo vision systems as reliably as tools dedicated specifically to this task.

Non-overlapping vision system. To demonstrate the ability of our approach to calibrate complex vision systems without overlapping, we rigidly fix four Realsense D415i cameras on a bar such that each camera looks in a different direction without any overlap between the stereo views, as depicted in Fig. 10(a). Since each RGB-D vision system is composed of two NIR cameras, a total of 8 cameras are being calibrated. To achieve this calibration, we use 4 boards, as shown in Fig. 10(c). The calibration result is available in Fig. 10(b), the mean reprojection error for the entire sequence of 1200 frames (per camera) is 0.19 pixel suggesting a very accurate estimation of the parameters of this vision rig. We additionally provide an all-around 3D reconstruction obtained from this calibrated system in Fig. 11.

Overlapping multi-camera system. To allow comparison against other multi-camera calibration toolbox (Rehder et al., 2016), we acquire a calibration sequence (~ 900 frames) from 6 cameras sharing a significant overlapping FoV (see Fig. 12). In Fig. 13 we propose a comparison of the estimated parameters between our approach and Kalibr (Rehder et al., 2016). We can

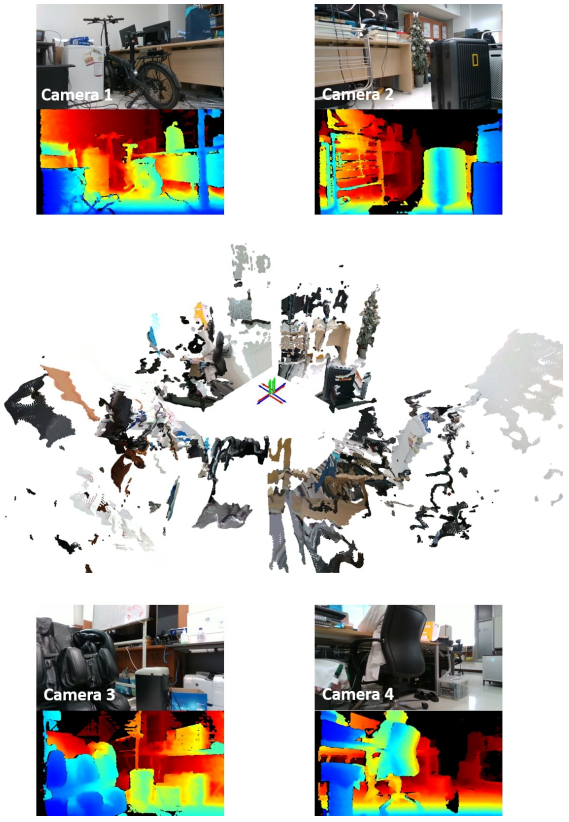


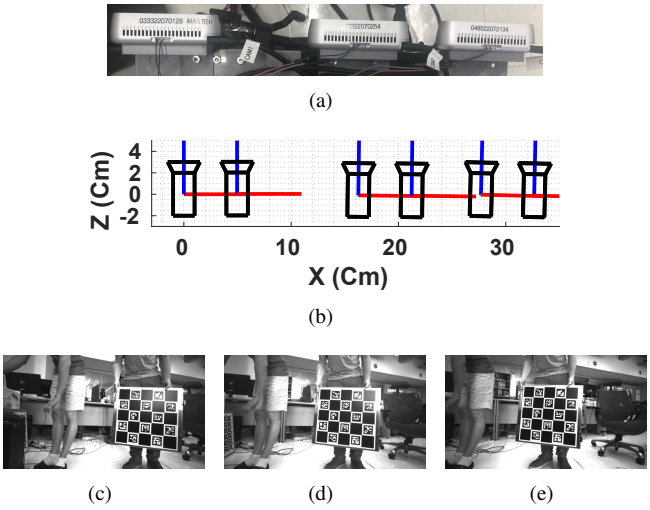
Fig. 11. All around reconstruction from 4 non-overlaping RGB-D cameras.

notice that both toolboxes lead to relatively similar solutions with a maximum rotational difference under 0.2° and less than 3mm difference in the translation estimation. We can notice that the difference in the estimation increases with the camera index which theoretically might be related to the behavior of Kalibr that computes the camera poses pairwise leading to a potential drift when many cameras are being calibrated. Also, it should be noticed that the calibration using Kalibr (Rehder et al., 2016) took more than 1 hour while our calibration technique achieves parameters estimation in less than 4 minutes. Using our estimated parameters, we have aligned the point clouds obtained by the 3D sensors to observe the quality of the overlap between the views (see Fig. 12 (b)). We can notice that the alignment of the 3D reconstructions is accurate and also validate our methodology for such type of multi-camera systems.

Hybrid stereo-vision system. While the previous experiments exclusively focused on homogeneous vision systems – composed of similar types of camera, MC-Calib can also deal with hybrid vision systems composed of perspective and fisheye cameras. Therefore, we propose a setup composed of one fisheye and one perspective camera, as depicted in Fig. 5.2(a). To calibrate this system we used a single calibration board. The resulting calibration lead to a mean reprojection error of 0.15px and the estimated baseline is 19.8cm which is consistent with the organization of the cameras in the rig. We additionally propose a stereo rectification result (see Fig. 5.2(b)) which also confirm the quality of our calibration.

Table 3. Stereo calibration parameters comparison between our approach and Bouguet (2004). f_L, f_R, pp_L and pp_R are the focal lengths and principal points of the left and right cameras respectively.

Methods \ Parameters	f_L (px)	f_R (px)	pp_L (px)	pp_R (px)	XYZ Rotation Euler (°)	XYZ Translation (cm)
Bouguet (2004)	703.81	707.12	(631.27, 372.15)	(650.37, 386.23)	(−0.05, 0.61, −0.47)	(−11.98, 0.02, −0.07)
Ours	701.52	704.33	(629.06, 375.08)	(651.27, 387.25)	(−0.0214, 0.3731, −0.5487)	(−12.0231, 0.0250, −0.1151)
Difference	2.29	2.79	3.67	1.35	0.25	0.05



(a) (b) (c) (d) (e)

Fig. 12. Calibration of 6 overlapping cameras. (a) Multi-camera rig, (b) calibration result, (c-d) three images used for this calibration.

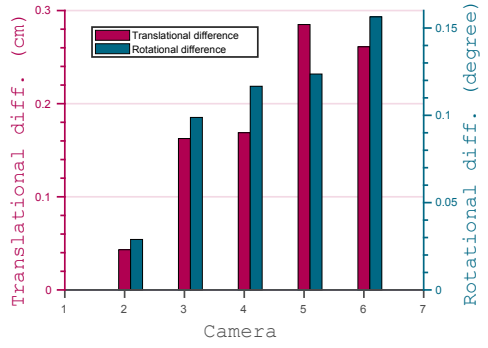
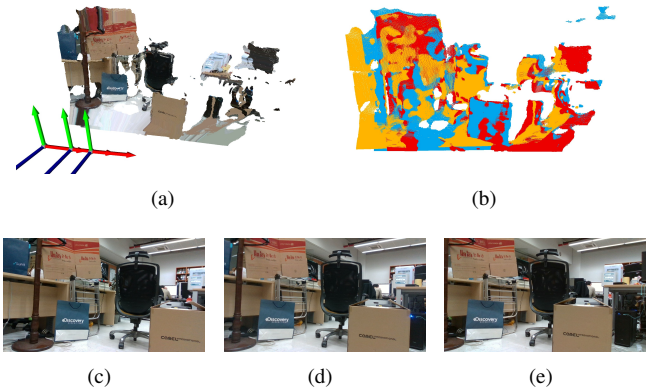


Fig. 13. Translational and rotational difference between ours and Kalibr (Rehder et al., 2016) for the calibration of 6 overlapping cameras.

Converging vision system. The final real setup we propose to explore is a converging multi-camera system composed of 4 pairs of stereo NIR cameras, as depicted in Fig. 16. The system is calibrated with a cube of 30cm side on which each face is covered with a unique calibration pattern. The geometry of this cube is unknown for this calibration and any 3D object composed of planes would also be applicable for this calibration. The calibration outputs include the intrinsic/extrinsic parameters of the cameras and the geometry of the calibration object (see the top of Fig. 16). This calibration is with 1500 images for each of the 8 cameras in the system and the mean reprojection error is 0.28px. Qualitatively, our calibration result is consistent with the experimental setup. However, since no ground truth is available, we propose to reconstruct a 3D object by combining



(a) (b) (c) (d) (e)

Fig. 14. 3D reconstruction from 3 calibrated RGB-D cameras. (a) Merged point cloud from the 3 RGB-D reconstructions, (b) overlay of the registered point cloud where the yellow, blue and red depicts the cameras 1,2,3 respectively, (c-e) color image from each camera.

the four 3D point clouds captured by the RGB-D cameras to better highlight the accuracy of our method. This reconstruction is visible in Fig. 5.2. We can notice that this reconstruction is consistent, suggesting an accurate calibration of our system.

5.3. Robustness against noise

In this test, we would like to assess the robustness of the proposed technique against noise on corners' location. For this purpose, we simulate (numerically on Matlab) three sets of ideal camera systems with known parameters, motions, and 3D boards location. The first setup consists of a stereo vision system calibrated with three calibration boards attached together. The second simulated use case is a small light-field setup composed of a 3×3 camera matrix looking at 9 calibration boards. The final setup is a non-overlapping camera system made of two back-to-back stereo vision systems each looking at a grid of 9 checkerboards. In all of these experiments, the synthetic cameras are assumed to have a resolution of 1824×1376px and each board contains 6×6 corners. The rig's motions have been generated randomly in a given range of rotation and translation. To test the robustness of the calibration pipeline, Gaussian noise is added to the corners' positions. Different noise levels are used and 100 trials are performed for each noise level. The obtained results are visible in Fig. 18. Note that in this experiment, the RANSAC threshold has been intentionally set higher than the maximum noise standard deviation to demonstrate our method's robustness using a set of noisy points.

Our system demonstrates good robustness to noise and, while the overall accuracy is impacted by the noise, no failure cases are observed over thousands of runs. Moreover, despite a

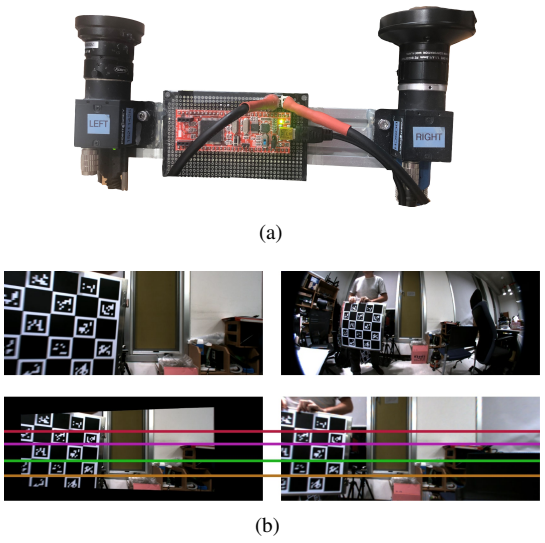


Fig. 15. Hybrid stereo-vision system calibration. (a) Picture of the system, (b, first row) perspective and fisheye images acquired by the cameras, (b, second row) rectified images with multiple epipolar lines displayed in color.

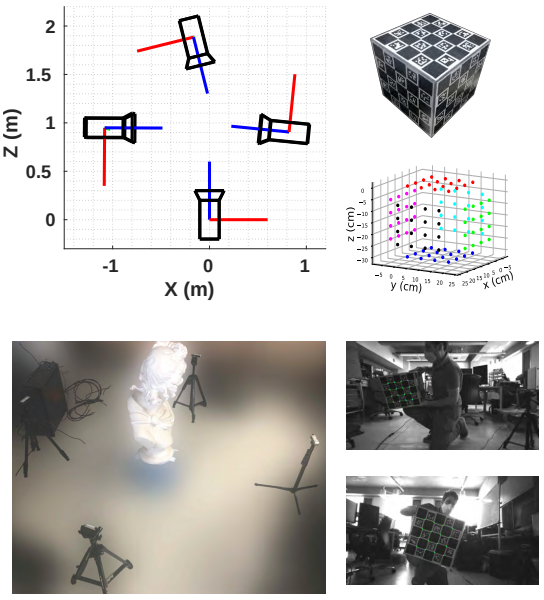


Fig. 16. Experimental setup and calibration results for a converging multi-camera system. (top-left) Calibration result (note that only the left camera of each RGB-D camera is displayed for clarity), (top-right) 3D calibration cube and its reconstruction, (bottom-left) experimental setup with a 3D object placed in the center of the camera, (bottom-right) images with detected corners from the camera 1 and 3 respectively.

very high noise, the rotation and translation error never exceed 2° and 0.08 meters regardless of the cameras' configuration. Worth noting that owing to the filtering and the corner refinement processes, in practice, it is very unlikely to reach inaccurate corner localization. Regarding the intrinsic parameters, the deviation from the ground truth remains reasonable with a maximum of 70px for the principal point and 80px for the focal length. Noticeably, the *light-field* configuration has higher intrinsic parameters errors which can be related to the limited variety of viewpoints in the sequence. Interestingly, the non-overlapping scenario, which is assumed to be more complex to resolve, seems to reach higher accuracy. This can be attributed to the robustness of the proposed bootstrapping strategy.

5.4. Robustness against outliers

This section confirms the stability of our approach against outliers. Following the same evaluation environments as in Sec. 5.3, we evaluate the robustness of our approach under the presence of hard outliers which may occasionally occur during the detection of fiducial markers. In contrast to Sec. 5.3, for this assessment, no noise is added to the inlier points. We compute the success rate (mean reprojection error inferior to 5px) over 100 trials for a different level of outlier contamination ranging from 0 to 70%. An outlier is a point with a deviation of at least more than 10px from its real pixel position (the outliers are generated randomly in the image with a uniform distribution). The same number of outliers is enforced for each image. The computed success rate is available in Fig. 19(a) where, in most scenarios, our solution is robust up to 60% of outlier contamination, leaving only 14 points per board to perform the calibration of the system. This resilience can be mostly attributed to the RANSAC algorithm used to reject incorrect points. Since the boards contain a relatively low number of points, 1000 RANSAC iterations are usually enough to discover a set of uncorrupted points to perform the system calibration. At a level of 70% of outliers, the success rate falls to zero for all three studied scenarios. In practice, such an extreme presence of outliers is highly unlikely. Nevertheless, without our RANSAC filtering and robust optimization scheme, even a very few outliers lead to a complete failure of the calibration.

5.5. Robustness evaluation of the hand-eye calibration

In this section, we highlight the relevance of our bootstrapped hand-eye calibration technique (covered in Sec. 4.6.2) for non-overlapping vision systems under the presence of wrongly estimated poses (outliers). Not only our technique allows a fast and constant time hand-eye calibration, but it is also significantly more robust to outliers thanks to our minibatch estimation and our rotation consistency testing stage. To evaluate the level of robustness offered by our framework, we synthetically generate 100 pairs of poses from two non-overlapping cameras. In

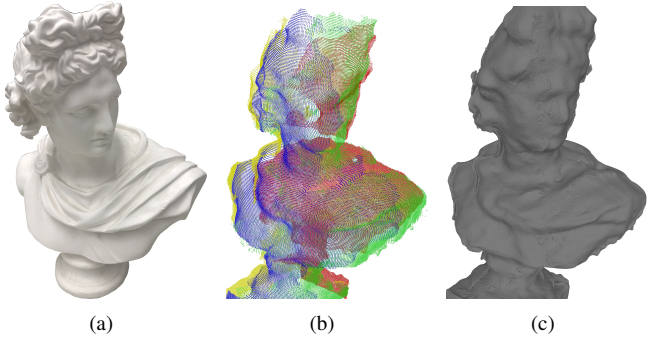


Fig. 17. Reconstructed object using our calibration parameters. (a) Picture of the object, (b) aligned 3D points cloud from the 4 RGB-D cameras displayed in red, green, blue, and yellow respectively, (c) meshed result.

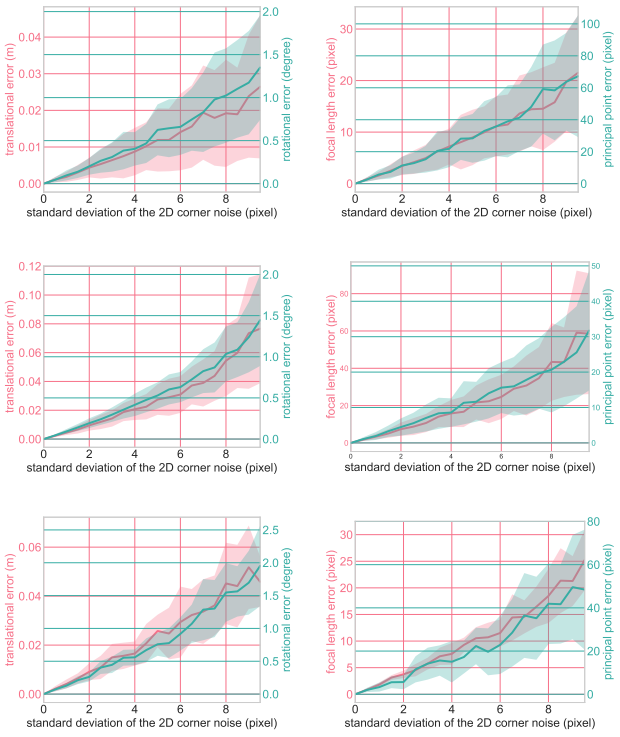


Fig. 18. Robustness against various quantity of noise with 100 iterations per level of noise, the tick lines represent the mean error value and the transparent envelopes depict the standard deviation. (first column) Translation and rotation error assessment for the three sequences: stereo, light-field, and non-overlapping, (second column) focal length and principal point error for the three sequences: stereo, light-field, and non-overlapping.

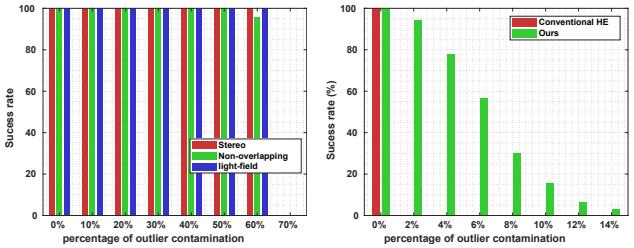


Fig. 19. Robustness against outliers. (a) Success rate of the entire calibration pipeline versus various outlier percentage contamination (for 3 scenarios depicted in red, green and blue) (b) success rate of our hand-eye calibration technique for various levels of outliers contamination (the red and green bars depict standard (Tsai et al., 1989) and our hand-eye calibration procedure respectively).

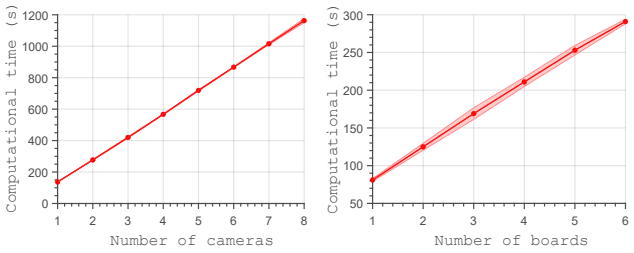


Fig. 20. (a) Computational time vs number of cameras (8 cameras, 1200 images per camera and 4 calibration boards). (b) Computational time vs number of boards (2 cameras, 550 images, 6 calibration boards). The transparent red envelop depicts the standard deviation.

Fig. 19(b), we provide the success rate for 500 trials at a different level of outlier corruption. In this experiment, we do not include any non-linear refinement of the pose. We consider the pose estimation successful if the accuracy in rotation and translation are lower than 5° and 2cm respectively.

To better understand the importance of the proposed technique, we compare our solution with a standard hand-eye calibration solution that directly tries to resolve the problem using all the available poses (Tsai et al., 1989). These conventional hand-eye calibration techniques have not been designed to deal with outliers, thus, the presence of a single outlier leads to very large errors as underlined in Fig. 19(b). On the contrary, our technique is specifically designed to deal with outliers and allows estimating the inter-camera pose even if multiple outliers are contaminating the set (with 6% of outliers our algorithm can return a successful pose estimation 60% of the time).

5.6. Computational time

While our strategy has not been deliberately designed to be computationally effective – since the calibration stage is usually conducted offline, our C++ implementation allows a relatively quick calibration of any camera system. To give an overview of the method's speed, we have performed tests with various number of cameras (see Fig. 20(a)) and boards (see Fig. 20(b)). In these experiments, the calibration is repeated 25 times for each instance to analyze the mean and standard deviation of the elapsed time. To test with a various number of cameras, we examined a non-overlapping system composed of 4 stereo cameras as described in Sec. 5.2. In this experiment, 1200 images per camera are captured, raising the total number of images to be processed to 9600, while 4 boards are utilized. In this context, it can take up to 15 minutes to calibrate the entire system. However, the computational time decreases significantly when reducing the number of cameras utilized. To evaluate the computational time versus the number of employed boards, we use a stereo sequence of 550 images of a calibration cube composed of 6 boards. We decrease the number of boards and measure the computational time for each scenario ranging from 1 to 6 boards. Once again, we can notice (see Fig. 20(b)) that the calibration time decreases with fewer boards. The reason is that a larger number of boards to be detected also leads to more computation for their detection.

To better understand which part of the algorithm is time-consuming, we propose to analyze the mean computational time per stage of the algorithm (see Table 4). This evaluation is achieved with 4 non-overlapping stereo cameras (1200 frames per camera and 4 calibration boards). As can be seen, the most time-consuming part is the detection of the Charuco boards followed by the initialization of the intrinsic parameters. The rest of the proposed calibration process is very light and takes under one minute to calibrate complex multi-camera systems.

Table 4. Computational time per stage for 4 non-overlapping stereo systems (8 cameras) with 1200 frames per camera and 4 boards.

Stage \ Time	(s)	(%)
Boards detection	1049.4	90.8
Intrinsic estimation	92.0	7.9
Objects merging	1.4	0.1
Camera merging	2.2	0.19
Non-overlap. calib.	6.6	0.6
Final Optimization	4.2	0.3
Total	1155.9	100

6. Conclusion

In this paper, we have presented one of the most flexible, robust, and user-friendly camera calibration toolbox to date. It allows calibrating fisheye, perspective, and hybrid vision systems composed of an arbitrary number of cameras without any priors or restrictions on their location. Moreover, an arbitrary number of calibration boards can be used and placed without specific limitations. Regarding the stability of the technique, our hierarchical calibration strategy ensures a good convergence of the intrinsic and extrinsic parameters of the camera rig. This architecture combines robust estimation strategies (i.e. bootstrapped initialized of non-overlapping, RANSAC, and robust non-linear optimization) to ensure a satisfying calibration. Through a large series of experiments, we have demonstrated the robustness, accuracy, and relevance of the approach for multiple use cases. Our toolbox still has a few limitations. In its current form, it can only exploit Charuco markers while the addition of more advanced AR markers, such as AprilTag (Olson, 2011), might be an interesting extension. Besides, additional camera models could be included, such as spherical camera models (Usenko et al., 2018; Barreto, 2006). Finally, MC-Calib is designed for unsynchronized camera systems or for rolling shutter cameras. Aside from these restrictions, our technique does not suffer other limitations other than usual corner detection-related problems (e.g. motion blur, out-of-focus blur, etc.). We believe this work can be useful for most applications requiring multi-camera systems, in particular in robotics and for autonomous cars where multiple fisheye cameras are often employed.

Acknowledgement

Francois Rameau was supported under the framework of international cooperation program managed by the National Research Foundation of Korea(NRF-2020M3H8A1115028, FY2022). Jinsun Park was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1I1A1A01060267).

References

Alexiadis, D.S., Chatzitofis, A., Zioulis, N., Zoidi, O., Louizis, G., Zarpalas, D., Daras, P., 2016. An integrated platform for live 3d human reconstruction and motion capturing. IEEE Transactions on Circuits and Systems for Video Technology (TCSVT) 27, 798–813.

Ataer-Cansizoglu, E., Taguchi, Y., Ramalingam, S., Miki, Y., 2014. Calibration of non-overlapping cameras using an external slam system, in: International Conference on 3D Vision.

Barreto, J.P., 2006. A unifying geometric representation for central projection systems. Computer Vision and Image Understanding (CVIU) .

Bouguet, J.Y., 2004. Camera calibration toolbox for matlab. http://www. vision. caltech. edu/bouguetj/calib_doc/index. html .

Caron, G., Eynard, D., 2011. Multiple camera types simultaneous stereo calibration, in: ICRA.

Community, B.O., 2018. Blender - a 3D modelling and rendering package. Blender Foundation. Stichting Blender Foundation, Amsterdam.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numerische mathematik 1, 269–271.

Duane, C.B., 1971. Close-range camera calibration. Photogramm. Eng 37, 855–866.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395.

Forbes, K., Voigt, A., Bodika, N., 2002. An inexpensive, automatic and accurate camera calibration method, in: South African Workshop on Pattern Recognition.

Gao, X.S., Hou, X.R., Tang, J., Cheng, H.F., 2003. Complete solution classification for the perspective-three-point problem. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 25, 930–943.

Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F.J., Marín-Jiménez, M.J., 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognition 47, 2280–2292.

Ha, H., Perdoch, M., Alismail, H., So Kweon, I., Sheikh, Y., 2017. Deltille grids for geometric camera calibration, in: ICCV.

Heng, L., Choi, B., Cui, Z., Geppert, M., Hu, S., Kuan, B., Liu, P., Nguyen, R., Yeo, Y.C., Geiger, A., et al., 2019. Project autovision: Localization and 3d scene perception for an autonomous vehicle with a multi-camera system, in: ICRA.

Heng, L., Li, B., Pollefeys, M., 2013. Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry, in: IROS.

Im, S., Ha, H., Rameau, F., Jeon, H.G., Choe, G., Kweon, I.S., 2016. All-around depth from small motion with a spherical panoramic camera, in: ECCV.

Itseez, 2015. Open source computer vision library.

Kannala, J., Brandt, S.S., 2006. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. PAMI 28, 1335–1340.

Keselman, L., Iselin Woodfill, J., Grunnet-Jepsen, A., Bhowmik, A., 2017. Intel realsense stereoscopic depth cameras, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops.

Kumar, R.K., Ilie, A., Frahm, J.M., Pollefeys, M., 2008. Simple calibration of non-overlapping cameras with a mirror, in: CVPR.

Kuo, J., Muglikar, M., Zhang, Z., Scaramuzza, D., 2020. Redesigning slam for arbitrary multi-camera systems, in: ICRA.

Lébraly, P., Deymier, C., Ait-Aider, O., Royer, E., Dhome, M., 2010. Flexible extrinsic calibration of non-overlapping cameras using a planar mirror: Application to vision-based robotics, in: IROS.

Lébraly13, P., Ait-Aider13, O., Royer23, E., Dhome13, M., 2010. Calibration of non-overlapping cameras-application to vision-based robotics, in: BMVC.

Li, B., Heng, L., Koser, K., Pollefeys, M., 2013. A multiple-camera system calibration toolbox using a feature descriptor-based calibration pattern, in: IROS.

Lin, Y., Larsson, V., Geppert, M., Kukelova, Z., Pollefeys, M., Sattler, T., 2020. Infrastructure-based multi-camera calibration using radial projections, in: ECCV.

Liu, A., Marschner, S., Snavely, N., 2016. Caliber: Camera localization and calibration using rigidity constraints. International Journal of Computer Vision (IJCV) 118, 1–21.

Lloyd, S., 1982. Least squares quantization in pcm. IEEE transactions on information theory 28, 129–137.

Mei, C., Rives, P., 2007. Single view point omnidirectional camera calibration from planar grids, in: ICRA.

Moulon, P., Monasse, P., Perrot, R., Marlet, R., 2016. Openmvg: Open multiple view geometry, in: International Workshop on Reproducible Research in

Pattern Recognition.
Munoz-Salinas, R., 2012. Aruco: a minimal library for augmented reality applications based on opencv. Universidad de Córdoba 386.
Mur-Artal, R., Tardós, J.D., 2017. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics (TRO) 33, 1255–1262.
Olson, E., 2011. Apriltag: A robust and flexible visual fiducial system, in: ICRA.
Pesce, M., Galantucci, L., Percoco, G., Lavecchia, F., 2015. A low-cost multi camera 3d scanning system for quality measurement of non-static subjects. Procedia CIRP 28, 88–93.
Qin, T., Li, P., Shen, S., 2018. Vins-mono: A robust and versatile monocular visual-inertial state estimator. IEEE Transactions on Robotics (TRO) 34, 1004–1020.
Rameau, F., Demonceaux, C., Sidibé, D., Fofi, D., 2014. Control of a ptz camera in a hybrid vision system, in: VISAPP.
Rehder, J., Nikolic, J., Schneider, T., Hinzmann, T., Siegwart, R., 2016. Extending kalibr: Calibrating the extrinsics of multiple imus and of individual axes, in: ICRA.
Rosinol, A., Abate, M., Chang, Y., Carlone, L., 2020. Kimera: an open-source library for real-time metric-semantic localization and mapping, in: ICRA.
Scaramuzza, D., Martinelli, A., Siegwart, R., 2006. A toolbox for easily calibrating omnidirectional cameras, in: IROS.
Schönberger, J.L., Frahm, J.M., 2016. Structure-from-motion revisited, in: CVPR.
Schroers, C., Bazin, J.C., Sorkine-Hornung, A., 2018. An omnistereoscopic video pipeline for capture and display of real-world vr. ACM Transactions on Graphics (TOG) 37, 1–13.
Strauß, T., Ziegler, J., Beck, J., 2014. Calibrating multiple cameras with non-overlapping views using coded checkerboard targets, in: 17th international IEEE conference on intelligent transportation systems (ITSC), IEEE. pp. 2623–2628.
Sturm, P., Ramalingam, S., 2011. Camera models and fundamental concepts used in geometric computer vision. Now Publishers Inc.
Tsai, R., 1987. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal on Robotics and Automation 3, 323–344.
Tsai, R.Y., Lenz, R.K., et al., 1989. A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration. IEEE Transactions on robotics and automation 5, 345–358.
Urban, S., Wursthorn, S., Leitloff, J., Hinz, S., 2016. MultiCol Bundle Adjustment: A Generic Method for Pose Estimation, Simultaneous Self-Calibration and Reconstruction for Arbitrary Multi-Camera Systems. International Journal of Computer Vision (IJCV) , 1–19.
Usenko, V., Demmel, N., Cremers, D., 2018. The double sphere camera model, in: 3DV.
Wang, J., Olson, E., 2016. Apriltag 2: Efficient and robust fiducial detection, in: IROS.
Wu, C., et al., 2011. Visualsfm: A visual structure from motion system .
Xing, Z., Yu, J., Ma, Y., 2017. A new calibration technique for multi-camera systems of limited overlapping field-of-views, in: IROS.
Yu, Z., Yoon, J.S., Lee, I.K., Venkatesh, P., Park, J., Yu, J., Park, H.S., 2020. Humbi: A large multiview dataset of human body expressions, in: CVPR.
Zhang, Z., 2000. A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 22, 1330.
Zhao, F., Tamaki, T., Kurita, T., Raytchev, B., Kaneda, K., 2018. Marker-based non-overlapping camera calibration methods with additional support camera views. Image and Vision Computing 70, 46–54.