Flexible Calibration of Color and Depth Camera Arrays

Razmik Avetisyan   Christian Rosenke   Oliver Staadt

Visual Computing Lab, Institute for Computer Science
University of Rostock
18059 Rostock, Germany

{razmik.avetisyan2, christian.rosenke, oliver.staadt}@uni-rostock.de

ABSTRACT

In this work we present a flexible approach for calibrating an array of multiple stationary color and depth cameras using an optical tracking system. Our application domain is focused on 3D telepresence. Calibrating cameras in this area is still a major problem due to the limited applicability of common calibration approaches. Usually, groups of cameras are calibrated relative to each other by either requiring heavily overlapping fields of view for many pairs of participating cameras or free movable cameras.

Our method moves away from these techniques by calibrating every camera individually. The key technology applied is a tracked calibration target with permanently identified global location provided by a tracking device. Detecting the known target geometry in a camera image provides, beside intrinsic calibration parameters, the position of the camera relative to the target. Combining these two aspects of the calibration target's location makes it possible to register every camera in the common tracking coordinate system. We validate our approach using our prototype with 12 Firewire color cameras, 3 Kinect depth cameras, an OptiTrack tracking device, and a checkerboard with an attached trackable rigid body (see Figure 1). In this setup, we achieve a reprojection error of below 0.5 pixels on average.

Keywords
Camera Calibration, Registration, Camera Arrays, Telepresence

1 INTRODUCTION

Multi-camera acquisition setups combining color and depth cameras have become more and more popular in the recent years. A typical example is given by telepresence systems, where the user and the space around her have to be captured from an array of cameras [Maimone and Fuchs, 2011]. A common technical difficulty in the realization of telepresence systems and other multi-camera setups is their accurate calibration, desired at sub-pixel level. Without calibrated cameras, processing the parallelly produced imagery becomes much harder, if not impossible, especially in real time.

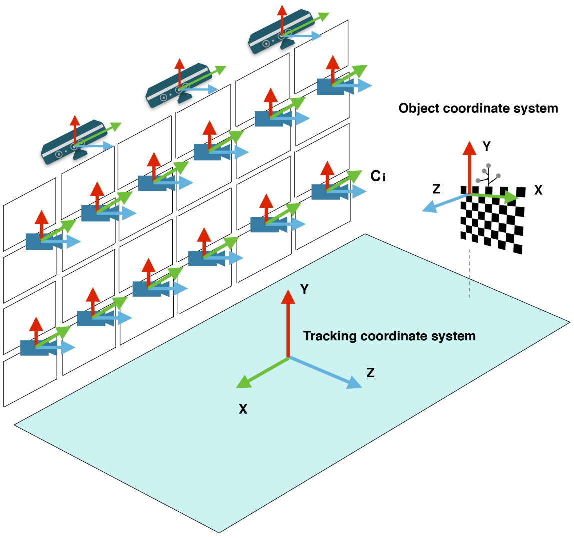

Figure 1: Schematic presentation of our telepresence prototype with 12 RGB and 3 RGB-D cameras integrated into the bezels of a large high-resolution display (LHRD).

The challenge of camera calibration is to find a set of internal and external parameters that describe the physical and geometrical characteristics of all involved cameras and their mutual relations. Intrinsic parameters of a camera, like focal length, principal point, and lens distortion coefficients, describe the non-linear behavior in the projection of scene objects onto the image plane. If known, these parameters can be used to correct the non-linear distortion in the recorded images such that they can be treated as if obtained by a pin hole camera. In addition, if the camera is a depth sensor, the intrinsic parameters should also describe the distortion of reported depth values to be able to correct them. The external calibration parameters, on the other hand, describe the camera positions and orientations. Basically, they can be given by a translation vector and a rotation matrix per camera to register them in a joint coordinate system.

Whereas intrinsic calibration of color cameras can be considered as a solved problem in the setting of telepresence, for instance by using a checkerboard and the OpenCV functions based on Zhang's method [Zhang and Zhang, 2000], obtaining extrinsic calibration parameters remains difficult.

Most extrinsic calibration methods available today (see for instance [Bouguet, 2004, Szeliski and Shum, 1997]) assume that many subsets of cameras, usually consisting of two cameras, have an essentially overlapping field of view. The geometrical relation between the cameras of one group is obtained by placing an object of known geometry, like a checkerboard, into the shared field of view. The global setup is subsequently obtained by fitting together the local parameters. However, joining the local parameters coming from different groups is a numerically involved optimization problem. Moreover, the small local errors in every parameter subset tend to add up into a considerable global error. Other extrinsic calibration schemes rely on conditions that are not satisfactory in a telepresence scenario, like free movable cameras, for instance.

To address the problem of calibrating multiple cameras with not necessarily overlapping fields of view, we extend our idea from [Avetisyan et al., 2014]. Similar to our previous work, we apply a tracking system to determine position and orientation of a calibration target in a global coordinate system. Although such tracking systems are quite expensive, they are usually part of telepresence systems to provide a natural interaction experience to users [Lehmann and Staadt, 2013]. In contrast to [Avetisyan et al., 2014], we use this reference data not only to perform a depth correction for cameras, but also to register every camera individually in the global coordinate system. Furthermore, we achieve higher accuracy by using a more robust arrangement of tracking markers. Basically, for every camera individually, the target is placed into the field of view and intrinsic parameters are obtained. Subsequently, the features of the target are detected in a sequence of intrinsically corrected images. Combining the known geometry, position and orientation of the target with the coordinates of the projected features determines position and orientation of the camera with respect to the global tracking coordinate system.

Our prototype setup includes 12 Firewire color cameras with a resolution of 1024×768, arranged in a planar configuration, three depth cameras with a resolution of 640×480 and a 6DOF tracking system (see Figure 1). We use a checkerboard with an attached trackable rigid body. In our setup we were able to accurately calibrate the cameras such that the reprojection error fell below 0.5 pixels on average. The approach described in this work can be easily integrated into a telepresence system, like the one presented in [Willert et al., 2010].

The remainder of this paper is organized as follows: In Section 2, the existing methods are summarized. Section 3 presents the proposed approach for calibrating cameras. In Section 4, we evaluate our approach and, finally, concluding remarks are given in Section 5.

2 RELATED WORK

Most state-of-the-art camera calibration approaches are based on calibration targets with known geometry such as a checkerboard (see for instance [Zhang and Zhang, 2000]) like in our case. Today, there are a number of according toolboxes available [Bouguet, 2004, Barreto et al., 2003, Geiger et al., 2012, Scaramuzza et al., 2006, Svoboda et al., 2005] that implement such an approach for the calibration of setups consisting of two or more cameras. However, most of these methods have the limitation that they make use of overlap in the fields of views for many pairs of participating cameras. Subsequently, we provide a short overview on techniques like these, which are basically distinguished by the specific calibration target that has to be observed in all captured images of a specific camera subgroup. In fact, the simplest possible calibration target geometry is provided by a single light point, which is used in [Barreto et al., 2003, Svoboda et al., 2005]. In [Christoph et al., 2011] the authors show that, for cameras with overlapping fields of view, an active calibration target, like a display showing a temporally varying pattern, can provide better results than a static one. The work [Li et al., 2013], on the other hand, proposes a static pattern that contains much more features than others. In this way only a small fraction of the target has to be visible within each camera image. In [Fernández-Moral et al., 2014] a target is not explicitly required as, instead, they use an overlap of features in the surrounding planar environment, like walls and the ceiling, for instance. If depth cameras are contained in the setup, specialized calibration targets are required. In [Kainz et al., 2012] the authors describe a method for multiple Kinect cameras, where, to obtain extrinsic parameters, they use a target with Kinect-visible markers that are simultaneously detected by a number of these cameras. In [Teng et al., 2014], the authors firstly compute local calibration parameters to register the color and depth streams of each Kinect camera and then they interpolate these values across the entire captured volume for registering the cameras relative to each other.

However, in general, telepresence systems do not guarantee overlap in the fields of view. Regardless of the used target, this often makes the above standard approach to multiple camera calibration inappropriate for the considered application. Therefore, to compensate the lack of overlap, several calibration methods apply mirrors [Agrawal, 2013, Hesch et al., 2010, Kumar et al., 2008, Lebraly et al., 2010, Sturm and Bonfort, 2006, Takahashi et al., 2012]. If the target cannot be brought into the field of view of a certain camera then a mirror is used to show at least the target's reflection. However, these techniques tend to be inaccurate in large setups like ours. The problem is that the distance between target and camera grows also with the reflection. Thus, the target becomes too small when seen in a mirror. Moreover, methods like these intensify the whole problem as they also have to determine the positions of the mirrors.

For settings, where cameras have insufficient overlap in there fields view, some other approaches rely on the portability of the cameras. For instance, the setup in [Caspi and Irani, 2001] uses the common motion of two closely bonded cameras over time to recover their geometrical relation. Beside the requirement to freely move the two cameras in space, they should also have the same center of projection. A similar approach for rigidly coupled but movable cameras is given in [Esquivel et al., 2007] using structure and motion techniques. Likewise, [Besl and McKay, 1992] proposes to calibrate depth cameras by moving the whole setup around. Here, a geometric iterative closest point method is used to register 3D points obtained from consecutive depth images. Another category of movable camera calibration is to estimate relative motion by odometry, as for instance in [Brookshire and Teller, 2012, Carrera et al., 2011, Heng et al., 2014, Heng et al., 2013, Lébraly et al., 2010, Schneider et al., 2013]. These methods apply to steadily moving cameras rigidly attached to some vehicle. In telepresence setup, however, cameras are rather rigidly connected to the whole setup and cannot be moved at all. Hence, we cannot take these methods into consideration.

A way of solving the problem with not sufficiently overlapping fields of views, which is more relevant to our application in telepresence, is to utilize the motion of objects in the scene [Makris et al., 2004, Micusik, 2011, Pflugfelder and Bischof, 2010, Radke, 2010, Rahimi et al., 2004, Tieu et al., 2005]. The idea is to not only fix the geometry of the target in advance but also the movement of the target. Based on that prior knowledge a camera can determine its own position by recognizing both, location and time of the target in the recorded images. However, all known methods require some calibration parameters to be given in addition. For example, the authors of [Pflugfelder and Bischof, 2010] assume the intrinsic parameters and the rotation of the cameras to be given in advance. Similarly, the authors of [Micusik, 2011] require the gravity vector directions for each camera to be able to estimate extrinsic parameters from target movements.

Another interesting approach, which has been developed for settings as ours, that is, large scale setup where cameras have rather different fields of view, is presented in [Ataer-Cansizoglu et al., 2014]. The authors scan the collaboration space with an external mobile device (SLAM system), like a simple depth camera, to get a 3D model of the acquisition setup. Afterwards they capture the scene with all cameras of the setup and fit the 2D image data onto the 3D capture. Then the 2D/3D correspondences between the stationary camera images and the 3D scan is used to locate every camera. The main problem with this approach is, that it heavily relies on the quality of depth images, which is known to be unstable and often inaccurate. Furthermore, finding point correspondences between a camera image and the entire 3D model is not straightforward and sometimes fails to work well.

The work [Beck et al., 2013] presents a full telepresence system that builds on up to three Kinect cameras per communication side. Although this system does not completely fit to our needs, as we also have color cameras in our setup, their calibration method represents a first starting point for our work. In particular, a milestone for calibration in this article is the presented way to correct depth values of a Kinect by the use of a tracking system. In their approach they determine the spacial relation of a depth camera that is mounted on a motorized platform and the static planar floor. Using this data as ground truth, they build a 3D lookup table that maps reported depth values to the associated positions in physical space. Recently, we simplified this method [Avetisyan et al., 2014] by replacing the complex motorized setup with a simple trackable checkerboard. The target is placed at various positions in physical space to fill the 3D lookup table of corrected depth values. Then, the recorded pairs of reported distance values from the camera and the 3D position of the target are used to interpolate a complete 3D lookup table with corrected depth values. In [Beck and Froehlich, 2015], the methods from [Beck et al., 2013] and [Avetisyan et al., 2014] are combined and refined, especially for the interpolation of corrected depth values.

Another interesting aspect of camera calibration in [Beck et al., 2013], which motivated us to advance their approach, is the method of extrinsic calibration of depth cameras. Like in our case, their technique applies a tracked calibration target to determine the position and orientation of a camera in two steps, firstly by the relation between camera and target and secondly by the known location of the target within the tracking volume. However, their target, a large box-shaped object, is only applicable for depth cameras. Yet, they suffer from problems with detecting the target in the noisy depth image as well as some numerical trouble also caused by the intense noise. In this paper, we present an enhanced, yet much simpler and faster, way to calibrate arbitrary stationary color and depth camera setups that eliminates the problems caused by the needs for overlapping views, mirrors, movable cameras, or complex error-prone numerical computations.

3 PROPOSED METHOD

In our setup we require that the space in front of the cameras is entirely observed by a tracking device, as for instance a Natural Point OptiTrack device like in our case. Moreover, we need a calibration target that fulfills the following requirements: (1) The exact 3D location of all calibration target features can be obtained from the tracked position and orientation of the target. (2) There are target features that are visible in color cameras while others (or even the same) can be seen in infrared images as recorded by depth sensors. Subsection 3.3 proposes a method to create a trackable checkerboard by attaching a rigid body with tracking markers, such that the calibration features, that is, the corner points between black and white squares, are precisely registered in the local coordinate system of the board. Our checkerboard features are clearly visible in both color and depth cameras.

The following presents our calibration procedure together with the underlying basic ideas. To calibrate a (telepresence) system, we move the calibration target in front of each camera j as shown in the Figure 1 and detect the n 2D feature points

$$F_{ij} = \{(x_{ij1}, y_{ij1})^T, \ldots, (x_{ijn}, y_{ijn})^T\} \qquad (1)$$

of the given calibration target in a sequence of camera images taken at times i. In our particular case F_{ij} consists of the $n = 48$ 2D coordinates for projected corner points between black and white squares on our checkerboard. Along with every feature point set F_{ij}, we synchronously record the global coordinates (t_i, r_i) of the checkerboard, where t_i is a translation vector and r_i a rotation matrix specifying location and orientation of the checkerboard in tracking system coordinates, that is, global coordinates. For every depth camera we vary the distance in our movement and also take a longer sequence to later be able to compute the intrinsic parameters according to [Avetisyan et al., 2014].

3.1 Intrinsic Calibration

With the 2D feature points F_{ij}, we are able to compute intrinsic camera parameters for all cameras. As we use a checkerboard in our experimental setting, we use the Open-CV standard method for intrinsic parameters that is based on [Zhang and Zhang, 2000]. Subsequently we can remove any recorded non-linear distortion in our camera images and at the same time, we obtain corrected 2D feature points F'_{ij}.

The intrinsic parameters for the correction of depth values are obtained by applying our approach from [Avetisyan et al., 2014]. Result of this method is a 3D lookup table that maps every triple (x, y, d) of reported depth value d at pixel (x, y) to a corrected depth value d'. We use the corrected feature points F'_{ij} in combination with the global coordinates (t_i, r_i) and the known geometry of the calibration target to correct the reported depth values. In fact, for every time i, a detected feature point (x, y) in depth camera j comes with a recorded depth value d. Simultaneously, it corresponds to a 3D-point p given by the known geometry and the recorded position and orientation (t_i, r_i) of the target at time i. Accordingly, we can define the value in the lookup table at (x, y, d) to be $d' = \|p - (x, y, 0)^T\|$. As we record sufficiently long feature sequence, we have enough defined entries in the look up table to correctly interpolate values for empty spots. See [Avetisyan et al., 2014] for all the details.

3.2 Extrinsic Calibration

The estimation of extrinsic camera parameters is performed for every camera j individually. In contrast to other approaches, we do not use intersecting fields of view, similarities in motion of multiple cameras, or any other shared information. Instead, we apply the known geometry and coordinates of calibration target features and the actually recorded and corrected features F'_{ij} of camera j at the times i. We begin by computing the coordinates of the target relative to camera j for all times i. In other words, we obtain translation vectors T_{ij} and rotation matrices R_{ij} relative to the coordinate system of camera j. For our particular setting, where we use a checkerboard for a target, we again apply the OpenCV method based on [Zhang and Zhang, 2000] to obtain T_{ij} and R_{ij}.

The set of tuples $(t_i, r_i, T_{ij}, R_{ij})$ over all times i and all cameras j contains all information necessary to compute the extrinsic calibration parameters of the whole system. For every i and j we first calculate the position P_{ij} of camera j in the checkerboard coordinate system at time i:

$$P_{ij} = -R_{ij}^T \times T_{ij} \qquad (2)$$

Then we compute for all times i the translation vector τ_{ij} and the rotation matrix ρ_{ij} of every camera j relative to the global coordinate system as follows:

$$\tau_{ij} \quad = r_i \times P_{ij} + t_i \tag{3}$$
$$\rho_{ij} \quad = r_i \times R_{ij} \tag{4}$$

As a result, we obtain for every camera j a sequence of pairs (ρ_{ij}, τ_{ij}), each specifying position and orientation of camera j in the global coordinate system. In a perfect setting, all elements of this sequence would be the same. However, due to unstable environmental conditions, the calibration result varies over time i and it is up to us, to filter out outliers and faulty values. To determine the extrinsic calibration parameters of each camera j, our idea is to select the pair (ρ_{ij}, τ_{ij}) from the given sequence that minimizes the reprojection error, a widely accepted measure for calibration quality.

To determine the reprojection error for a given pair, we use our knowledge about the calibration target geometry together with the information about its location to compute for every time i a virtual representation of the target. In particular, we have a set V of 3D points representing the features of our virtual target. For the checkerboard, that we use in our experimental setup, V consists of the corner points between black and white squares on a 3D model of our board. Next, we move the virtual features to the recorded position of the real target at a time i by transforming every point $p \in V$ into the global coordinate system using

$$q = p \times r_i + t_i. \tag{5}$$

By that we get a set of transformed feature points V_i representing the target in its location with respect to tracking system coordinates at time i. To estimate the quality of a given pair (ρ_{ij}, τ_{ij}), we transform the point set V_i to the respective coordinate system of camera j by evaluating

$$r = \rho_{ij}^{\mathsf{T}} \times (q - \tau_{ij}) \tag{6}$$

for all points $q \in V_i$ to obtain the virtual feature point set V_{ij} in camera coordinates. Next, we use the intrinsics of camera j to project V_{ij} to the image plane of camera j and by that obtain a set

$$V'_{ij} = \{(X_{ij1}, Y_{ij1})^T, \ldots, (X_{ijn}, Y_{ijn})^T\} \tag{7}$$

of reprojected 2D feature points. Finally, we measure the reprojection error at time i and camera j by

$$\delta_{ij} = \sqrt{n^{-1} \sum_{k=1}^{n} (x_{ijk} - X_{ijk})^2 + (y_{ijk} - Y_{ijk})^2}, \tag{8}$$

the square root of the mean squared error. The pair (ρ_{ij}, τ_{ij}) that minimizes δ_{ij} is returned as the extrinsic calibration result for camera j.

Hence, as a result of the whole procedure we get a set $\{(R_1, T_1), (R_2, T_2), \ldots\}$ of pairs (R_j, T_j) specifying for every camera j the location T_j and the orientation R_j with respect to the global tracking system coordinate system. Clearly, this also yields the pairwise relation of all cameras which is usually determined in classical calibration approaches.

3.3 A Trackable Calibration Target for Color and Depth Cameras

For our calibration method, we have to create a trackable target that has features for both color and depth cameras. Instead of using a target that actually has geometric features that can be detected in depth images, like applied for instance in [Beck et al., 2013], we use a simple checkerboard. Then we detect the corners between black and white squares in the infrared image just as for ordinary color images. Solely the fact that the noisy infrared image is improved by a 5 × 5 median filter stands as a difference to the procedure for color images.

We create a tracked target by attaching a trackable rigid body with a static configuration of tracking markers on the top of a checkerboard similar to the one used in [Avetisyan et al., 2014]. Although this makes it possible to precisely track the position of the rigid body, the exact relation of these locations to the checkerboard features remains vague. To solve this problem, we attach four additional tracking markers onto the corners of the checkers field on the board. For a better understanding see Figure 2. Next, in an initialization step, we align the rigid body with the checkerboard target making use of the four newly attached markers. For this purpose we first create a virtual marker at the crossing of the diagonals given by the corner markers using the tracking system software. In this way, we exactly get the center of the checkerboard. Then we calculate the

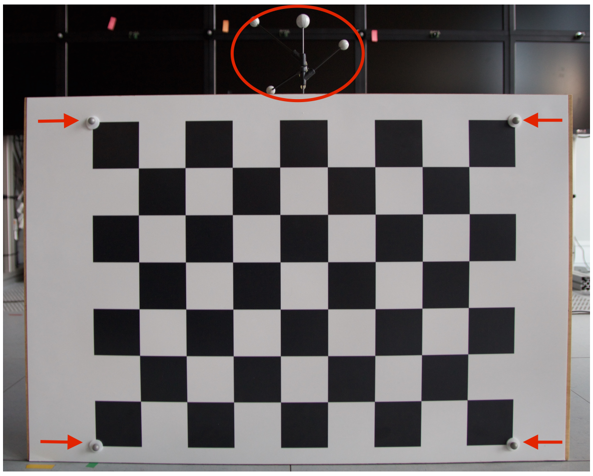

Figure 2: A checkerboard target with attached rigid body and four additional markers on the field corners.

offset from the center to the rigid body's coordinates. Figure 3 illustrates the described procedure. Finally, we use the rotation reported from the tracking system and the dimensions of the checkerboard recorded in the initialization to translate coordinates of the rigid body to the left top corner of the checkerboard, which is considered as the origin of our checkerboard coordinate system.

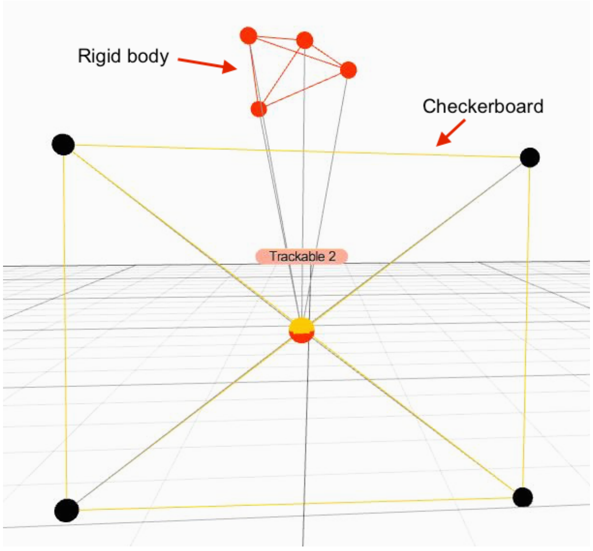

Figure 3: Geometric alignment between the rigid body and the checkerboard.

4 EVALUATION

The quality of our calibration method is highly dependent on the accuracy and calibration quality of the used tracking system. For our experiments we used a Natural Point OptiTrack optical tracking system. Twelve infrared cameras surround the calibration space, each of them delivering images with a maximum latency of 10 ms at sub-pixel image accuracy. With our current calibration, the system has around 0.145 mm mean error. We may safely assume that the influence of this error is by magnitudes of order below that of other error sources, like for instance the noise and sampling based inaccuracy in feature detection.

The few linear equations solved in our approach are numerically stable enough to be irrelevant for error estimation. In the following we demonstrate experimentally that other possible error sources have only a very small impact, too.

To evaluate our results quantitatively, we start with the widely accepted reprojection error, which is convenient as it is already implemented and used by our approach. Figure 4 shows an example for error estimation in a arbitrarily selected color image captured by a camera of our setup. For this particular measurement, we observed sub-pixel accuracy of around 0.45 pixels. We also tested different images captured with other cameras and we can report that with our method we are able to achieve reprojection errors permanently smaller than 0.5 pixels. It is worth to mention, that the calibration accuracy varies with the place where the checkerboard was positioned for the capture. In some regions of the tracking volume the tracking system's cameras have a better view on the target and then we observe reprojection errors of up to 0.1 pixels.

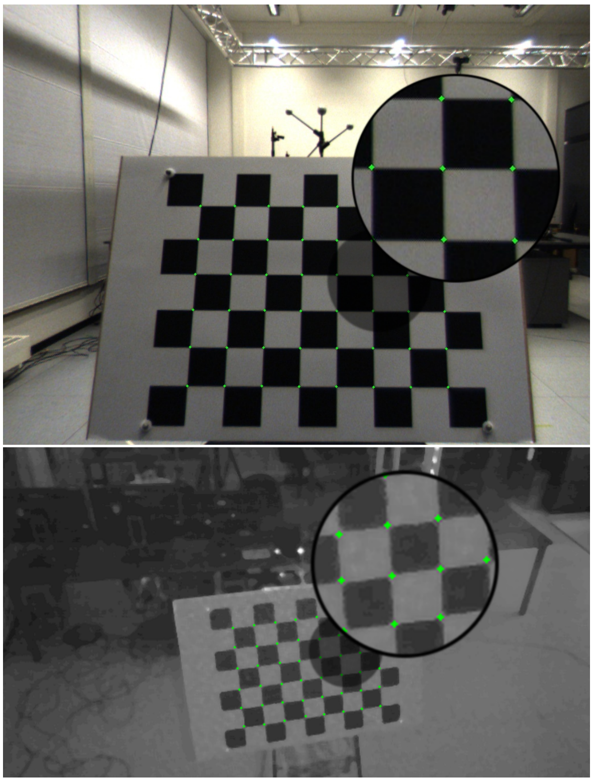

Figure 4: The reprojection error for color images (top) and infrared images (bottom). The reprojected points are shown using green points. Apperently, they are located very closely to the corner points on the checkerboard. See the magnified areas for more details.

The estimation of the reprojection error, as given in this section, states that every camera is accurately registered in the global coordinate system with a similar insignificant error. It follows that also the spatial relation between pairs of cameras is accurately determined as shown in the following experiment where we evaluate the mutual reprojection error between one pair of cameras c_1 and c_2. We place our checkerboard target in the shared field of view of the two cameras and, like in Section 3.1, detect and correct the 2D feature point sets F_1' and F_2' in both camera images. Similar to Section 3.2, using the OpenCV method based on [Zhang and Zhang, 2000] for F_1' provides a translation T_1 and a rotation R_1 specifying the location of the checkerboard relative to camera c_1. Using the spatial relation between c_1 and c_2, which was calculated by our calibration approach, we can transform (T_1, R_1) to (T_2, R_2) giving the location of the board, as seen by c_1, relative to camera c_2. Then we use the intrinsics of camera c_2 to project the virtual features to the image plane of c_2 and, like in Equation 7, obtain a set V_1' of reprojected 2D feature points. Finally, the reprojection error δ between V_1' and F_2' is obtained as in Equation 8.

Using this method we observed a mutual reprojection error of less than 0.5 pixels. It is worth to mention for our method that, as every camera is individually registered in the global coordinate system, the calibration error in the spatial relation between two cameras does not depend on the actual choice of the cameras or the fact that they have intersecting fields of view. Only for the estimation of the mutual reprojection error, our experiment needed clearly intersecting fields of view.

5 CONCLUSION

We have presented an extremely simple method for reliably calibrating multiple cameras using a tracking system with a trackable calibration target. Our approach does not utilize any further mutual information among neighboring cameras and enables us to calibrate cameras independently from each other in a fast and accurate fashion. Although the proposed method is very flexible and allows to calibrate dense camera arrays, its main weakness comes from the potentially high costs of installing a tracking system in a multi-camera setup, if not yet present.

6 REFERENCES

[Agrawal, 2013] Agrawal, A. K. (2013). Extrinsic camera calibration without a direct view using spherical mirror. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 2368–2375.

[Ataer-Cansizoglu et al., 2014] Ataer-Cansizoglu, E., Taguchi, Y., Ramalingam, S., and Miki, Y. (2014). Calibration of non-overlapping cameras using an external slam system. In *3D Vision (3DV), 2014 2nd International Conference on*, volume 1, pages 509–516.

[Avetisyan et al., 2014] Avetisyan, R., Willert, M., Ohl, S., and Staadt, O. (2014). Calibration of depth camera arrays. In *Proceedings of SIGRAD 2014, Visual Computing, June 12-13, 2014, Gothenburg, Sweden*, pages 41–48.

[Barreto et al., 2003] Barreto, J. P., Daniilidis, K., Kelshikar, N., Molana, R., and Zabulis, X. (2003). Easycal camera calibration toolbox.

[Beck and Froehlich, 2015] Beck, S. and Froehlich, B. (2015). Volumetric calibration and registration of multiple rgbd-sensors into a joint coordinate system. In *3D User Interfaces (3DUI), 2015 IEEE Symposium on*, pages 89–96.

[Beck et al., 2013] Beck, S., Kunert, A., Kulik, A., and Froehlich, B. (2013). Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):616–625.

[Besl and McKay, 1992] Besl, P. and McKay, N. D. (1992). A method for registration of 3-d shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(2):239–256.

[Bouguet, 2004] Bouguet, J.-Y. (2004). Camera calibration toolbox for matlab.

[Brookshire and Teller, 2012] Brookshire, J. and Teller, S. J. (2012). Extrinsic calibration from per-sensor egomotion. In *Robotics: Science and Systems VIII, University of Sydney, Sydney, NSW, Australia.*

[Carrera et al., 2011] Carrera, G., Angeli, A., and Davison, A. (2011). Slam-based automatic extrinsic calibration of a multi-camera rig. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2652–2659.

[Caspi and Irani, 2001] Caspi, Y. and Irani, M. (2001). Alignment of non-overlapping sequences. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 76–83 vol.2.

[Christoph et al., 2011] Christoph, S., Frank, F., and Elli, A. (2011). Camera calibration: active versus passive targets. *Optical Engineering*, 50(11).

[Esquivel et al., 2007] Esquivel, S., Woelk, F., and Koch, R. (2007). Calibration of a multi-camera rig from non-overlapping views. In Hamprecht, F., Schnörr, C., and Jähne, B., editors, *Pattern Recognition*, volume 4713 of *Lecture Notes in Computer Science*, pages 82–91. Springer Berlin Heidelberg.

[Fernández-Moral et al., 2014] Fernández-Moral, E., Jiménez, J. G., Rives, P., and Arévalo, V. (2014). Extrinsic calibration of a set of range cameras in 5 seconds without pattern. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, September 14-18, 2014*, pages 429–435.

[Geiger et al., 2012] Geiger, A., Moosmann, F., Car, O., and Schuster, B. (2012). Automatic camera and range sensor calibration using a single shot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3936–3943.

[Heng et al., 2014] Heng, L., Burki, M., Lee, G. H., Furgale, P. T., Siegwart, R., and Pollefeys, M. (2014). Infrastructure-based calibration of a multi-camera rig. In *2014 IEEE International Conference on Robotics and Automation 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 4912–4919.

[Heng et al., 2013] Heng, L., Li, B., and Pollefeys, M. (2013). Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 1793–1800. IEEE.

[Hesch et al., 2010] Hesch, J. A., Mourikis, A. I., and Roumeliotis, S. I. (2010). Extrinsic camera calibration using multiple reflections. In *Computer Vision - ECCV 2010, 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV*, pages 311–325.

[Kainz et al., 2012] Kainz, B., Hauswiesner, S., Reitmayr, G., Steinberger, M., Grasset, R., Gruber, L., Veas, E., Kalkofen, D., Seichter, H., and Schmalstieg, D. (2012). Omnikinect: Real-time dense volumetric data acquisition and applications. In *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*, VRST '12, pages 25–32, New York, NY, USA. ACM.

[Kumar et al., 2008] Kumar, R., Ilie, A., Frahm, J.-M., and Pollefeys, M. (2008). Simple calibration of non-overlapping cameras with a mirror. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7.

[Lebraly et al., 2010] Lebraly, P., Deymier, C., Ait-Aider, O., Royer, E., and Dhome, M. (2010). Flexible extrinsic calibration of non-overlapping cameras using a planar mirror: Application to vision-based robotics. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 5640–5647.

[Lébraly et al., 2010] Lébraly, P., Royer, E., Ait-Aider, O., and Dhome, M. (2010). Calibration of non-overlapping cameras - application to vision-based robotics. In *Proceedings of the British Machine Vision Conference*, pages 10.1–10.12. BMVA Press. doi:10.5244/C.24.10.

[Lehmann and Staadt, 2013] Lehmann, A. and Staadt, O. (2013). Distance-aware bimanual interaction for large high-resolution displays. In Csurka, G., Kraus, M., Laramee, R., Richard, P., and Braz, J., editors, *Computer Vision, Imaging and Computer Graphics. Theory and Application*, volume 359 of *Communications in Computer and Information Science*, pages 97–111. Springer Berlin Heidelberg.

[Li et al., 2013] Li, B., Heng, L., Koser, K., and Pollefeys, M. (2013). A multiple-camera system calibration toolbox using a feature descriptor-based calibration pattern. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 1301–1307.

[Maimone and Fuchs, 2011] Maimone, A. and Fuchs, H. (2011). Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, pages 137–146.

[Makris et al., 2004] Makris, D., Ellis, T., and Black, J. (2004). Bridging the gaps between cameras. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–205–II–210 Vol.2.

[Micusik, 2011] Micusik, B. (2011). Relative pose problem for non-overlapping surveillance cameras with known gravity vector. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3105–3112.

[Pflugfelder and Bischof, 2010] Pflugfelder, R. and Bischof, H. (2010). Localization and trajectory reconstruction in surveillance cameras with nonoverlapping views. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(4):709–721.

[Radke, 2010] Radke, R. (2010). A survey of distributed computer vision algorithms. In Nakashima, H., Aghajan, H., and Augusto, J., editors, *Handbook of Ambient Intelligence and Smart Environments*, pages 35–55. Springer US.

[Rahimi et al., 2004] Rahimi, A., Dunagan, B., and Darrell, T. (2004). Simultaneous calibration and tracking with a network of non-overlapping sensors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–187–I–194 Vol.1.

[Scaramuzza et al., 2006] Scaramuzza, D., Martinelli, A., and Siegwart, R. (2006). A toolbox for easily calibrating omnidirectional cameras. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5695–5701.

[Schneider et al., 2013] Schneider, S., Luettel, T., and Wuensche, H.-J. (2013). Odometry-based online extrinsic sensor calibration. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 1287–1292.

[Sturm and Bonfort, 2006] Sturm, P. and Bonfort, T. (2006). How to compute the pose of an object without a direct view? In Narayanan, P., Nayar, S., and Shum, H.-Y., editors, *Computer Vision – ACCV 2006*, volume 3852 of *Lecture Notes in Computer Science*, pages 21–31. Springer Berlin Heidelberg.

[Svoboda et al., 2005] Svoboda, T., Martinec, D., and Pajdla, T. (2005). A convenient multicamera self-calibration for virtual environments. *Presence: Teleoperators and Virtual Environments*, 14(4):407–422.

[Szeliski and Shum, 1997] Szeliski, R. and Shum, H.-Y. (1997). Creating full view panoramic image mosaics and environment maps. In *Computer Graphics (SIGGRAPH'97 Proceedings)*, pages 251–258, Los Angeles. Association for Computing Machinery, Inc.

[Takahashi et al., 2012] Takahashi, K., Nobuhara, S., and Matsuyama, T. (2012). A new mirror-based extrinsic camera calibration using an orthogonality constraint. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1051–1058.

[Teng et al., 2014] Teng, D., Bazin, J.-C., Martin, T., Kuster, C., Cai, J., Popa, T., and Gross, M. (2014). Registration of multiple rgbd cameras via local rigid transformations. *IEEE International Conference on Multimedia & Expo*.

[Tieu et al., 2005] Tieu, K., Dalley, G., and Grimson, W. (2005). Inference of non-overlapping camera network topology by measuring statistical dependence. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1842–1849 Vol. 2.

[Willert et al., 2010] Willert, M., Ohl, S., Lehmann, A., and Staadt, O. G. (2010). The extended window metaphor for large high-resolution displays. In *Proceedings of the Joint Virtual Reality Conference of EGVE - EuroVR - VEC, Stuttgart, Germany, 2010.*, pages 69–76.

[Zhang and Zhang, 2000] Zhang, Z. and Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1330–1334.