SIGRAD 2014
M. Obaid, D. Sjölie, E. Sintorn and M. Fjeld (Editors)

Calibration of Depth Camera Arrays

Razmik Avetisyan, Malte Willert, Stephan Ohl and Oliver Staadt

Visual Computing Lab, University of Rostock, Germany

Abstract

Depth cameras are increasingly used for tasks such as 3-D reconstruction, user pose estimation, and human-computer interaction. Depth-camera systems comprising multiple depth sensors require careful calibration. In addition to conventional 2-D camera calibration, depth correction for each individual device is necessary. In this paper, we present a new way of solving the multi depth-camera calibration problem. Our main contribution is a novel depth correction approach which supports the generation of a 3-D lookup table by incorporating an optical marker-based tracking system. We verify our approach for the Microsoft Kinect and for the MESA Swiss-Ranger4000 time-of-flight camera.

Categories and Subject Descriptors (according to ACM CCS): I.4.1 [Image Processing and Computer Vision]: Digitization and Image Capture—Camera Calibration

1. Introduction

The popularity of multi-camera acquisition setups in combination with depth cameras brought up new challenges for camera calibration.

A system with only one camera usually has several drawbacks. Due to the limited field of view, a single camera is not suitable for capturing large scenes, and when using one camera image, the scene and objects are visible only from one side. Furthermore, scene reconstruction suffers from object occlusion problems. Incorporating multiple neighboring depth cameras can overcome these limitations by using images from different viewpoints. The matching correspondence between different depth images can be used to achieve higher accuracy in measurements. Moreover, a good calibration of the camera system is the key requirement to achieve a high correspondence between the cameras.

Despite the camera registration, there is also particular need to calibrate the depth reported by a sensor in order to attain more accurate results. In numerous experiments [BKKF13, MCAPL13, MBPF12, SSBH11] it was observed that low-cost commodity depth cameras suffer from measurement inaccuracies when capturing over larger distances. As a result, the recorded datasets from different cameras cannot be matched to each other properly, thus, introducing inaccuracies in the reconstructed results.

In this paper, we offer a new depth correction approach incorporating an optical tracking system. Unlike other approaches [BKKF13, MBPF12], based upon complex mechanical setups for moving the camera, our approach incorporates the benefits of a tracking system. This facilitates the camera calibration in place, which means, that the cameras can remain fixed in their desired setup during the entire calibration procedure. Thus, intrinsic and extrinsic parameters possibly estimated beforehand remain valid. Another important advantage of our method is, that it allows to perform the depth correction of multiple cameras at once.

2. Related Work

There are three different categories of the calibration: (1) the intrinsic and extrinsic calibration, (2) intrinsic depth calibration, and (3) extrinsic calibration between color and depth cameras. Intrinsic calibration refers to estimating the internal camera parameters, for instance focal lengths. External parameters, such as the location and orientation of the camera, are determined in the extrinsic calibration To fully solve the camera calibration problem these tasks have be combined.

Intrinsic and extrinsic calibration: The basic idea of all approaches is to find the parameters by the help of an object with known geometrical properties. A large number of

algorithms use a planar checkerboard target for the calibration. The approaches in [AZD13, HCKH12, MCAPL13, MBPF12, Wen12] for example use the checkerboard for the intrinsic calibration. However, homography computed from the checkerboard pattern provides very suitable constrains for both intrinsic and extrinsic calibration. Extrinsic calibration approaches based on a checkerboard target are presented in [MBPF12, Wen12]. Liu et al. [LFZL12] describe an intrinsic calibration based on a cross shaped checkerboard target. Here, the authors use the missing corners of the checkerboard to visualize the error. Stürmer et al. [SSBH11] present an approach for aligning multiple ToF cameras based on a cubic reference object of known size. With the knowledge about angular and distance relations of the cubic sides the transformations between those cameras are calculated. Alexiadis et al. [AZD13] use a 1-D target based extrinsic calibration approach. Here, the authors establish point correspondences across all cameras for performing a pairwise stereo calibration based on epipolar geometry. Macknojia et al. [MCAPL13] describe a plane feature-based extrinsic calibration approach. The method consists of finding a normal vector and the center of the target plane for estimating the relative orientation and translation between the cameras. Another plane-based method is presented by Auvinet et al. [AMM12]. Their procedure involves moving a large rectangle in front of two cameras simultaneously. The intersection from three detected planes in each camera view is then used for constructing a virtual point. Having a cloud of virtual points synchronized in all camera views, the authors calculate the external relations between the cameras. Beck et al. [BKKF13] describe an extrinsic calibration approach based on a tracked box as reference. The transformation of each camera is estimated with respect to the constructed coordinate system of the box, which is the result of intersecting the detected box planes in the depth image.

Intrinsic depth calibration: The depth values reported by depth cameras are usually inaccurate. To correct these distances several approaches use manually measured data.

Maimone et al. [MBPF12] present a depth calibration approach based on a laser range finder. The authors fix the camera on a sliding rail and move it towards a wall. For every fixed interval they measure the real distance from the camera to the wall with a laser range finder. These measured distances along with the camera recorded depth values are used to determine a linear depth correction function. A more precise depth correction approach based on a 3-D lookup table is proposed by Beck et al. [BKKF13]. For generating this kind of table, a motor equipment is used, which slowly lowers the camera setup from the ceiling of the room towards the floor. During this process the lookup table is filled with tracking system measured and camera captured distances to the even floor. The resulting 3-D table provides a quite accurate per-pixel and per-distance mapping at every pixel in the depth image. An approach that does not need a mechanical pre-processing is offered by Herrera et al. [HCKH12]. They get improved depth values by adding a special varying offset to the distorted disparity, which decays exponentially with the increase in disparity.

Extrinsic calibration between color and depth cameras: Color to depth calibration is a quite well known problem, where the external relation between color and depth cameras has to be found. Kreylos [Kre13] replaces the planar checkerboard with a semi-transparent one. For this kind of target, checkerboard corners become visible in the color and in the depth image, which allows him to calibrate both kinds of cameras simultaneously with respect to each other, without need of exchanging the target-type. Macknojia et al. [MCAPL13] capture a fixed checkerboard target by both the color and IR image of Kinect camera simultaneously. They use a classical calibration method proposed by Zhang [Zha00] for finding the external relations. Other approaches like [HCKH12, LFZL12] calibrate cameras with respect to the same reference frame independently and then relate them to each other.

3. Our Calibration Approach

Depth correction is the main challenge for today's calibration approaches. Most cases use a correction table based on manually measured data. The drawback of the existing approaches is their incapability to generate this table for the cameras fixed in the desired setup. Furthermore the process usually needs special, quite elaborate hardware.

For this reason, we propose a new approach for correcting the depth values of a depth measuring device. The basic idea is to incorporate a 6DOF marker-based tracking system for measuring distances. The whole calibration procedure is separated into three steps. First, we carry out our new method of depth correction. Second, checkerboard-based intrinsic and extrinsic calibration between color cameras is followed. Finally, we execute extrinsic calibration between the color and depth cameras.

3.1. Depth calibration

To acquire tracking data we use a rigid body attached to a traditional checkerboard target (Fig. 1).

During the calibration procedure the rigid body is located by a tracking system, while at the same time the checkerboard is located in the camera images. Combining these two kinds of information allows us to calibrate the camera.

For every position of the checkerboard in space the system is capable to deliver coordinates (X_O, Y_O, Z_O) and orientation R of the rigid body with respect to our tracking coordinate system. We align the rigid body with the checkerboard target in such a way that the position of the body exactly matches with the center of the checkerboard, and the rotation of it

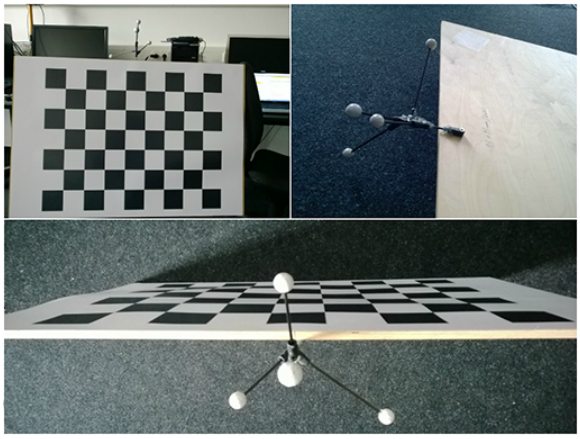

Figure 1: *The rigid body attached to the checkerboard.*

indicates the rotation of the checkerboard. To achieve this center adjustment, we use the tracking system's capability to specify offsets for the reported trackable coordinates.

While moving the target in space we can now take advantage of the reported 6DOF coordinates as follows:

We multiply the checkerboard center position with the rotation matrix R to rotate the checkerboard to its original orientation.

$$\begin{bmatrix} X_{R(O)} \\ Y_{R(O)} \\ Z_{R(O)} \end{bmatrix} = R \begin{bmatrix} X_O \\ Y_O \\ Z_O \end{bmatrix} \tag{1}$$

where X_O, Y_O, Z_O are the coordinates of the checkerboard center in the tracking coordinate system, R is the rotation matrix and $X_{R(O)}, Y_{R(O)}, Z_{R(O)}$ are the rotated coordinates of the checkerboard center. Having the coordinates of the rotated center, means having the checkerboard positioned orthogonal to the z-axis of the tracking coordinates system. By knowing the size of the edge of a checkerboard square it becomes possible to calculate 3-D coordinates $(X_{R(i)}, Y_{R(i)}, Z_{R(i)})$ of all inner corners by the following equations:

$$X_{R(i)} = X_{R(O)} + \left(i \; mod \; w - \left\lfloor \frac{w}{2} \right\rfloor\right) \alpha \tag{2}$$

$$Y_{R(i)} = Y_{R(O)} + \left(\left\lfloor \frac{h}{2} \right\rfloor - \frac{i}{w}\right) \alpha \tag{3}$$

$$Z_{R(i)} = Z_{R(O)} \tag{4}$$

where w, h are the number of inner corners in horizontal and vertical directions respectively, α is a constant indicating the size of the edge of a checkerboard square and $(X_{R(i)}, Y_{R(i)}, Z_{R(i)})$ are the coordinates of the i-th corner ($i = 0 \ldots [w \times h - 1]$). We count i starting from the top left corner of the checkerboard and continue row by row, column by column with an increment by one unit. Once the coordinates of the corners are calculated, we multiply all points with the inverse of the rotation matrix in order to obtain their real coordinates in the tracking coordinate system.

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = R^{-1} \begin{bmatrix} X_{R(i)} \\ Y_{R(i)} \\ Z_{R(i)} \end{bmatrix} \tag{5}$$

Since our tracking coordinate system does not coincide with the coordinate system of the camera, an extrinsic calibration has to be done to find the relation between those two coordinate systems. After obtaining this relation, we translate the 3-D positions of all checkerboard corners (X_i, Y_i, Z_i) to the camera coordinate system with the help of the following relation:

$$\begin{bmatrix} X_{K(i)} \\ Y_{K(i)} \\ Z_{K(i)} \end{bmatrix} = R_K \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + T_K \tag{6}$$

where R_K is the 3×3 rotation matrix and and T_K is the translation vector between those two coordinate systems respectively and $(X_{K(i)}, Y_{K(i)}, Z_{K(i)})$ are translated and rotated coordinates of the checkerboard corners. Having the coordinates of all checkerboard corners, we calculate the distances between the camera frame and each corner point according to the following equation:

$$D_{K(i)} = \sqrt{X_{K(i)}^2 + Y_{K(i)}^2 + Z_{K(i)}^2} \tag{7}$$

For each position of the checkerboard in space we have to measure $w \times h$ distance values. In general, this can be done easily, if a semi-transparent checkerboard is used. However, for the cameras used in this work, we provide a simpler approach that uses the IR images of the Kinect or SwissRanger4000 camera, respectively. We localize the inner corners of the checkerboard in the IR image in order to extract their positions in image coordinate space. As we know the relation between IR and depth cameras, we can map the corners onto the depth image in a quite accurate way.

After obtaining pairs of real measured distances and camera recorded distances, we generate a 3-D lookup table. The size of the table corresponds to the resolution of the depth image $T[resx, resy, raw]$ and the device depth sensitivity level as a third dimension. This makes it possible to map raw values to measured values for all pixels along the

depth image. To fill the lookup table we use the following equation:

$$T[x, y, depth(x, y)] = D_{K(i)} \tag{8}$$

where (x, y) are the pixel coordinates of i-th corner of checkerboard, $depth(x, y)$ is a depth measured by the camera and $D_{K(i)}$ is a distance of that point measured by a tracking system.

For each depth camera we construct a separate lookup table. To fill the lookup table and obtain precise depth mapping it is necessary to move the checkerboard in front of each camera with the objective to cover the whole volume in front of the device. Depending on the dimension of the depth image, the procedure may take several minutes to fill the lookup table with the real distance values. While moving the target in front of the camera, we make use of our visualization software providing real-time feedback, so that the user knows which parts of the volume are calibrated and which parts are not appropriately covered yet.

3.2. Intrinsic calibration and Extrinsic calibration

First, the intrinsic calibration procedure includes estimation of camera specific parameters for the RGB and the IR sensors. Such parameters include the focal length (f_x, f_y), the image center (u_0, v_0) and the lens distortion coefficients. To carry out this intrinsic calibration, we use a planar checkerboard pattern of a size 9×7. In average 15-20 images per camera were enough to ensure sufficient calibration quality. After obtaining the necessary amount of images with the checkerboard, we apply the standard calibration routine based on checkerboard homography. By doing so, the intrinsic parameters of the camera along with the distortion coefficients are estimated accurately.

Second, the procedure of extrinsic calibration involves positioning the checkerboard target in the overlapping region of two cameras. Having the target visible in both camera frames enables us to find the relative position and orientation for each of them. The standard routine for external calibration takes the already estimated intrinsic parameters along with the distortion coefficients to find the translation vector and rotation matrix for each camera pair.

3.3. Extrinsic calibration between color and depth cameras

Although the geometrical relation between the IR and RGB sensors of the Kinect is already embedded in the device internal memory by the manufacturer, it still may vary from one camera to another. For some tasks it is very important to have a more precise knowledge regarding to the external relation of these cameras. This is also true when we combine a depth camera, like the SwissRanger4000, with an external color camera, because in this setting we have no given knowledge about the geometrical relation. To find the relation, we fix the checkerboard target in front of the device in about 1.5 meters and capture it by both sensors. To enhance the detection rate for the IR image we apply a Median smoothing filter with the aperture size of 5 pixels.

After identifying the checkerboard corners on both images, we apply a standard routine for stereo calibration.

4. Prototype and Experiments

In the upcoming subsection a brief overview of hardware setup is presented. Then we discuss our prototype used for this work.

4.1. Experimental setup

The hardware setup of our system involves two kinds of depth cameras, the Kinect and the MESA SwissRanger4000, which are calibrated using 6DOF marker-based tracking system.

4.1.1. Depth cameras

The subject of our study consists of two Microsoft Kinect sensors and one MESA SwissRanger 4000 positioned in the room of a size about 4×4 meters. All cameras are positioned in such a way that they are pointing to the center of the working volume. The sensors are mounted and remain fixed during the entire calibration procedure. To avoid time and data synchronization issues over the network and make the data acquisition more simple we connect the depth cameras to a single portable PC.

4.1.2. Tracking system

For the depth correction we use a 6DOF marker-based tracking system from OptiTrack. Six V100:R2 cameras, which have a maximum latency of 10 ms, are surrounding the room and covering the calibration volume. The system is connected to the separate PC. The calibration of the tracking system is carried out beforehand, based on the routines provided by the manufacturer.

The two PCs used in the experiments are connected through a network. The tracking system machine streams the tracking data in real-time.

4.2. Prototype

Our prototype is a command line tool that combines the standard routines of OpenNI and OpenCV libraries for acquiring and processing RGB, depth and IR images in real-time. After collecting the necessary amount of images with a checkerboard target, we execute a standard routine for corner detection on both RGB and IR images. For IR images

the corner detection procedure is slightly unstable due to noise introduced by the IR projector. To overcome this problem, some authors suggest to block the IR projector with an overlapping mask and instead use powerful incandescent lamps [MCAPL13]. Another method for reducing the noise is to apply a smoothing filter [MCAPL13]. In our approach we apply a Median filter of a size of 5×5 pixels to reduce the noise and without blurring the images too much. To further enhance the detection rate we turned on the external IR lighting of the OptiTrack system. Fig. 2 illustrates enhanced IR image of a Kinect camera.

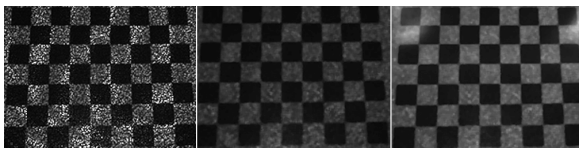

Figure 2: *The IR images of Kinect 1.) The noise IR image illuminated with IR projector 2.) Blurred IR image 3.) Blurred IR image with external IR lighting of the OptiTrack system.*

Since the SwissRanger 4000 has a relatively low resolution (176×144) compared to the Kinect, the standard checkerboard corner detection approach failed. This is due to the small size of the checkerboard visible in the amplitude image. Starting from 3 meters the detected corners were supposed to be very hard distinguishable, and from 3.5 meters onwards they were unrecognizable at all. To address this problem we scaled the amplitude images up to 3 times. Although the resulting images are a little bit blurry, the corner detection procedure works satisfying even on larger distances. Another problem with this device is the poor illumination of the target. For this reason, we increased the integration time of the camera up to 150 ms. This significantly enhancing the quality of images (Fig. 3), as the pixels could acquire more light.

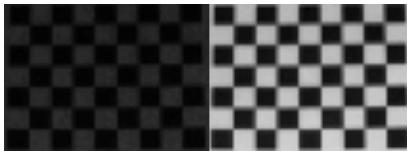

Figure 3: *The scaled amplitude image of SwissRanger4000. Left: Image captured with the default integration time. Right: Image captured with the integration time 150 ms.*

For our depth calibration procedure we make use the detected corners of the checkerboard in the IR image for finding corresponding corner coordinates in the depth image. According to Macknojia [MCAPL13] the offset between the IR image and the depth image are given by the following simple equations:

$$depth_x = ir_x + 5 \tag{9}$$

$$depth_y = ir_y + 4 \tag{10}$$

where ir_x and ir_y are the pixel coordinates in IR image, and $depth_x$ and $depth_y$ are the mapped pixel coordinates in depth image. The resulting images from Kinect and Mesa SwissRanger 4000 are presented in the Fig. 4 and Fig. 5 respectively.

Figure 4: *Left: The blurred IR image of Kinect camera. Right: Detected corners that are localized in the depth image.*

Figure 5: *Left: The amplitude image of SwissRanger 4000. Right: Detected corners that are localized in the depth image.*

4.3. Lookup table visualisation

Visualizing the 3-D depth lookup table is an essential step in the overall depth correction procedure. It allows the user to gain deeper knowledge regarding the status of the calibrated volume. Moreover, during the procedure of generating the table, the visualization tool may decrease the time needed for covering the whole volume by just providing a real-time feedback about which parts are filled and which parts still remain. This avoids of re-filling already covered parts.

To generate 3-D lookup table for the Kinect camera, we have to consider that the depth images of this device have 11 bits per pixel. Under the fact that one bit is reserved for invalid values only 10 bit remain. Hence, it is sufficient to have $2^{10} = 1024$ as the size of the third dimension. The resulting table has a size of $640 \times 480 \times 1024$. The presented lookup

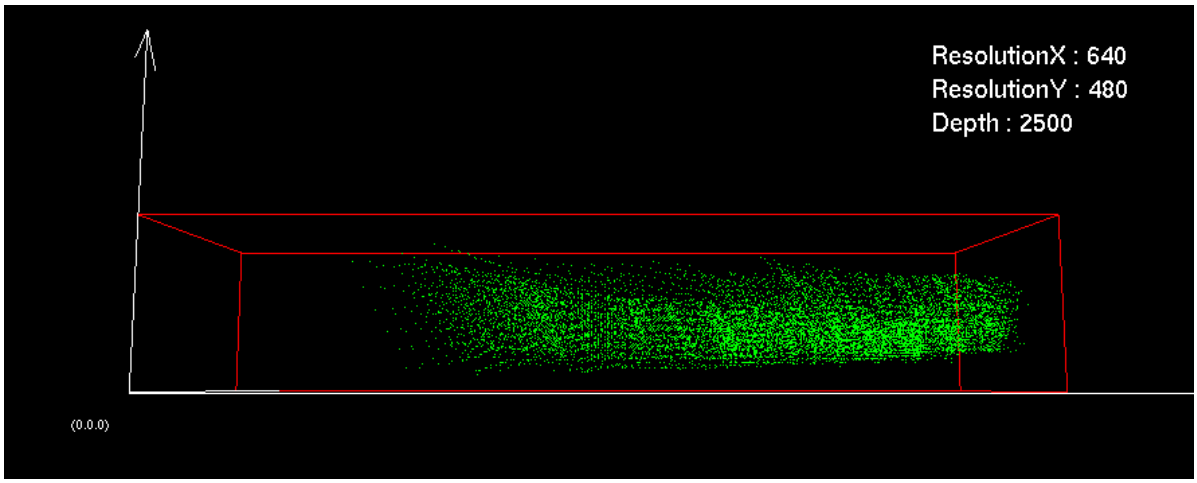

Figure 6: *3-D lookup table visualization.*

table can also be used for storing millimeter values instead of raw values, in the case when camera driver already returns converted values (in our experiments we stored millimeter values).

For the visualization of the table, we render a standard cube (see Fig. 6) and scale it according to the size of the lookup dimensions. The calibrated dataset is visualized inside of the table with a green color. The tool allows to rotate the cube and have a detailed view from all sides.

5. Results and Discussion

The entire procedure of our multiple depth camera calibration can be characterized as a multistage process, where each individual step has a significant influence on the subsequent ones. As already described, the depth correction is done according to our new developed concept that incorporates tracking data for generating 3-D lookup table. This offers superior depth camera calibration w.r.t previous approaches as the following argumentation explains:

As stated before, the existing depth calibration approaches [BKKF13, MBPF12] use laser rangefinders to measure distances. However, the distance is not measured for every pixel of the depth image but instead only for one point, usually the image center. Then, the distance for the rest of the pixels is just assumed to be the same, which introduces significant errors. See Figure 7 for clarification, where according to the described experimental setup, $distance1$ and $distance2$ are assumed to be equal.

To have a more precise depth mapping all possible distances between the camera and target have to be measured, which is accomplished for the first time in our approach.

We carried out experiments to demonstrate the estimated errors for the Kinect sensor. To do this, we positioned the

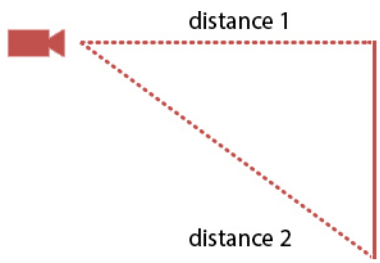

Figure 7: *Varying distance along the surface.*

checkerboard target in such a way, so that it is shifted from the camera center. As shown in the Figure 8, the target mostly occupies the left-bottom part of the depth image. This allows us to estimate the change of the depth error depending on the distance from the center.

For the corresponding depth errors see Table 1. The table has a size of $w \times h$ (number of inner corners of checkerboard) where every value in a cell represents the error for the corresponding checkerboard corner expressed in millimeters. At first glance, the relatively high error values become apparent, which increase with a growing distance to the image center. Notice, that the maximum error values are significantly higher then those observed in other papers.

To demonstrate the results of the proposed depth calibration approach, we visualize the dataset extracted from the surface of the checkerboard. In Fig. 9 the green dataset represents camera recorded depth values of checkerboard corners, whereas the yellow one corresponds to corrected depth values.

One advantage of our approach is that it doesn't require hard mechanical setup and equipment for measuring dis-

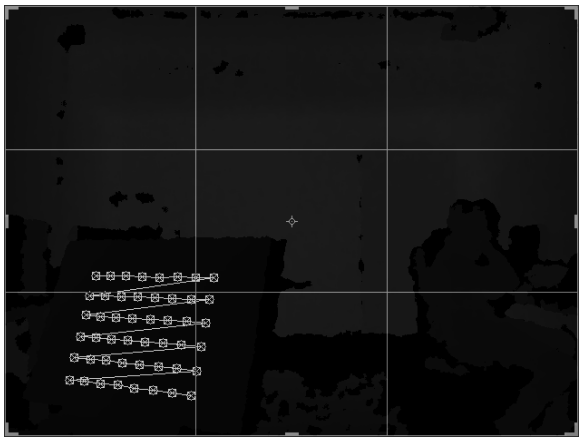

Figure 8: *Detected corners of the checkerboard positioned in about 2 meters from camera (target in the depth image mostly occupies the left-bottom part of the image, thus allowing to consider center shifted distances).*

74	65	56	48	21	13	-3	-9
86	76	76	47	38	29	20	12
100	99	87	76	56	46	37	27
126	114	102	89	77	66	63	53
153	131	118	113	109	96	84	81
183	151	145	132	126	113	115	110

Table 1: *Estimated depth errors measured for a single position of the checkerboard from Fig. 8 (all values are in **millimeters**).*

tances. The usage of a 6DOF marker-based tracking system allows to keep cameras fixed when caring out the camera calibration. In addition to this, our method supports the calibration of more then two cameras simultaneously, which reduces the required time.

In this work we also present a checkerboard-based 2-D calibration for a multi depth-camera system. For the two RGB sensors of the Kinect device we achieved a minimum re-projection error of about 0.3 pixel. This result is better then the original calibration provided by the manufacturer, which is greater than 0.5 pixel.

For an example, the tables presented below show estimated calibration parameters and distortion coefficients for two Kinect cameras $K1$ and $K2$.

Besides the intrinsic parameters, we also estimated the distortion coefficients in this step. In the tables presented below p_1, p_2 are the tangential and k_1, k_2, k_3 are the radial distortion coefficients respectively.

Extrinsic calibration results are presented in the Tables 6 and 7. The translation components are expressed in centimeters and rotation components are in Rodrigues representation.

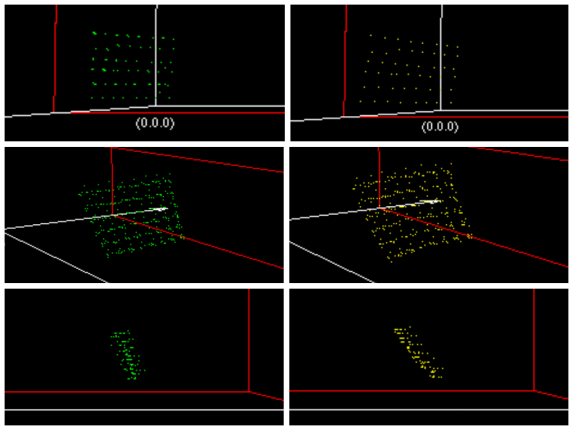

Figure 9: *Visualization of checkerboard corners. Left column: Corners estimated using camera depth values. Right column: Visualization of the same corners using corrected depth values.*

	f_x	f_y	u_0	v_0	*error*
K1	518.00	516.55	298.24	252.06	0.35
K2	523.08	522.35	312.70	248.28	0.31

Table 2: *Intrinsic calibration results of the RGB sensor.*

	f_x	f_y	u_0	v_0	*error*
K1	586.65	586.78	297.56	246.27	0.93
K2	581.53	580.81	292.65	246.36	0.94

Table 3: *Intrinsic calibration results of the IR sensor.*

	k_1	k_2	k_3	p_1	p_2
K1	0.191	-0.455	0.276	-0.001	-0.013
K2	0.247	-0.816	0.927	-0.009	-0.006

Table 4: *Distortion coefficients of the RGB sensor.*

	k_1	k_2	k_3	p_1	p_2
K1	-0.139	0.819	-2.964	-0.005	-0.024
K2	-0.236	1.654	-4.252	-0.007	-0.007

Table 5: *Distortion coefficients of the IR sensor.*

	T_x	T_y	T_z	R_x	R_y	R_z
K1	-18.0	15.5	258.5	-1.8	-2.2	0.4
K2	-58.3	7.2	132.4	1.6	1.4	5.7

Table 6: *Extrinsic calibration results for two Kinect sensors.*

	T_x	T_y	T_z	R_x	R_y	R_z
K1	2.47	-0.06	-0.68	-0.03	-0.03	0.00
K2	2.49	0.10	-0.34	-0.04	-0.04	0.00

Table 7: *Extrinsic calibration results between color and depth cameras of a Kinect sensor.*

6. Conclusion

In this paper we presented a novel depth correction approach incorporation an optical tracking system during the calibration process. Our method uses a 3-D lookup table to give a reliable per-pixel and per-distance mapping at every point in the depth image. In comparison to other methods, our approach support depth calibration of a camera in place using neither mechanical setups nor different types of distance measuring equipment that require the camera to be moved. Moreover, it facilitates the simultaneous depth correction of multiple cameras. To the best of our knowledge, there is no available depth correction algorithm capable to carry out the calibration of more than one camera at once. In addition to the proposed depth calibration approach, we also describe a standard checkerboard-based 2-D calibration. For acquiring the images we use the NITool. Being developed for this work the tool eases the data acquisition by enabling to switch from one camera to another and requesting necessary amount of frames per sensor. Additional support of parallel capturing improves the performance of the extrinsic calibration significantly.

Acknowledgements

This work was supported by the EU FP7 Marie Curie ITN "DIVA" under REA Grant Agreement No. 290227. We wish to thank the anonymous reviewers for their remarks. Special thanks to Christian Rosenke and Martin Luboschik for their valuable comments and suggestions.

References

[AMM12] AUVINET E., MEUNIER J., MULTON F.: Multiple depth cameras calibration and body volume reconstruction for gait analysis. In *Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on* (2012), pp. 478–483. 2

[AZD13] ALEXIADIS D., ZARPALAS D., DARAS P.: Real-time, full 3-d reconstruction of moving foreground objects from multiple consumer depth cameras. *Multimedia, IEEE Transactions on 15*, 2 (2013), 339–358. 2

[BKKF13] BECK S., KUNERT A., KULIK A., FROEHLICH B.: Immersive group-to-group telepresence. *Visualization and Computer Graphics, IEEE Transactions on 19*, 4 (2013), 616–625. 1, 2, 6

[HCKH12] HERRERA C. D., KANNALA J., HEIKKILÃĐ J.: Joint depth and color camera calibration with distortion correction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 34*, 10 (2012), 2058–2064. 2

[Kre13] KREYLOS O.: Kinect camera calibration, 2013. http://www.doc-ok.org/?p=289. 2

[LFZL12] LIU W., FAN Y., ZHONG Z., LEI T.: A new method for calibrating depth and color camera pair based on kinect. In *Audio, Language and Image Processing (ICALIP), 2012 International Conference on* (2012), pp. 212–217. 2

[MBPF12] MAIMONE A., BIDWELL J., PENG K., FUCHS H.: Enhanced personal autostereoscopic telepresence system using commodity depth cameras. *Computers and Graphics, Volume 36 Issue 7, November, 2012, Pages 791-807 36*, 7 (2012), 791–807. 1, 2, 6

[MCAPL13] MACKNOJIA R., CHAVEZ-ARAGON A., PAYEUR P., LAGANIERE R.: Calibration of a network of kinect sensors for robotic inspection over a large workspace. In *Robot Vision (WORV), 2013 IEEE Workshop on* (2013), pp. 184–190. 1, 2, 5

[SSBH11] STÜRMER M., SEILER C., BECKER G., HORNEGGER J.: Alignment of multiple Time-of-Flight 3D Cameras for Reconstruction of walking feet. In *ISB2011 Brussels, Conference book Program & Abstracts* (2011). 1, 2

[Wen12] WENDELIN K.-A.: *Combining Multiple Depth Cameras for Reconstruction*. Master's thesis, Institut für Softwaretechnik und Interaktive Systeme, Technische UniversitaÌĹt Wien, 2012. 2

[Zha00] ZHANG Z.: A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 22*, 11 (Nov 2000), 1330–1334. 2