A NOVEL KINECT V2 REGISTRATION METHOD FOR LARGE-DISPLACEMENT ENVIRONMENTS USING CAMERA AND SCENE CONSTRAINTS

Yuan Gao[†⋆], *Sandro Esquivel*[†], *Reinhard Koch*[†], *Matthias Ziegler*[⋆], *Frederik Zilly*[⋆], *Joachim Keinert*[⋆]

[†] Institute of Computer Science, Christian-Albrechts-University of Kiel, 24118 Kiel, Germany
{yga, sae, rk}@informatik.uni-kiel.de
[⋆] Fraunhofer Institute for Integrated Circuits IIS, 91058 Erlangen, Germany
{matthias.ziegler, frederik.zilly, joachim.keinert}@iis.fraunhofer.de

ABSTRACT

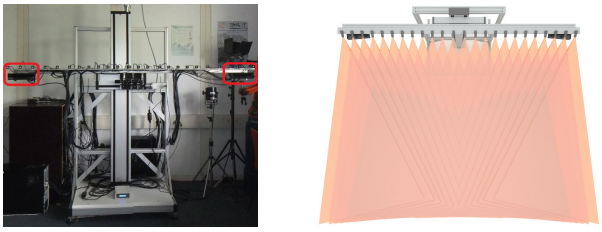
In a lot of multi-Kinect V2-based systems, the registration of these Kinect V2 sensors is an important step which directly affects the system precision. The coarse-to-fine method using calibration objects is an effective way to solve the Kinect V2 registration problem. However, for the registration of Kinect V2 cameras with large displacements, this kind of method may fail. To this end, a novel Kinect V2 registration method, which is also based on the coarse-to-fine framework, is proposed by using camera and scene constraints. Specifically, in the coarse estimation stage, scene constraints are explored using off-the-shelf feature point detectors and camera constraints are explored using homography and fundamental matrices. In the estimation refinement stage, an Iterative Closest Point (ICP)-based point cloud registration method is utilized. Experimental results show that the proposed Kinect V2 registration method using camera and scene constraints performs much better in precision than using calibration objects in the large-displacement environment.

Index Terms— Kinect V2 Registration, Large-Displacement Environment, Coarse-to-Fine Method, Fundamental Matrix, Iterative Closest Point

1. INTRODUCTION

The second version of the Microsoft Kinect (Kinect V2) is one of the most low-cost and high-speed Time-of-Flight (ToF) sensors in the market [1]. The comparison between Kinect V2 and the first generation of Microsoft Kinect (Kinect V1) is well studied in [2, 3, 4, 5], where Kinect V2 exhibits higher accuracy and better performance than Kinect V1 in multimedia applications. A more interesting advantage of Kinect V2 is the possibility of an interference-free multi-Kinect V2 setup. Currently, systems with multiple Kinect V2 sensors have attracted more and more research interests for their wide applications, *e. g.*, people tracking [6, 7], augmented reality [8] and motion capture [9].

Motivation: The multi-camera rig in Fig. 1 is a movable device for capturing dynamic light field built in the Multimedia Information Processing (MIP) laboratory of Kiel University [10]. Two Kinect V2 sensors are integrated in this system for the reason that the Field of View (FOV) of one Kinect V2 is too small compared with the large joint FOV of the other 24 RGB cameras, while utilizing two Kinect V2 can remedy this defect. Accurate registration or calibration of these two Kinect V2 cameras is very tough, considering the distance between these two Kinect V2 cameras is quite large, which is around **2.4** meters. Little literature focuses on this large-displacement depth sensor calibration problem and the traditional checkerboard-based calibration method [11] is prone to fail if the checkerboard is not huge enough for being captured by both cameras at the same time.



(a) Frontal view (b) Virtual vertical view

Fig. 1. The movable multi-camera rig. Red blocks indicate the positions of the Kinect V2 sensors.

Related work: As for multi-Kinect V2 calibration in non-large-displacement setups, several methods have been proposed. Palasek *et al*. leverage a checkerboard to calibrate two Kinect V2 cameras, where up to seven different views of this checkerboard should be captured to improve the calibration precision [12]. Beck *et al*. design an optical tracking system for immersive 3D telepresence which maps points in the depth images into a joint coordinate system with color information using a volumetric calibration and registration approach [13]. However, this approach heavily relies on the tracking of a continuously moving checkerboard. Munaro *et al*. develop a universal framework for scalable people tracking and calibration of different types of depth sensors, including Kinect V2, ToF and stereo cameras [6]. The people detection trajectories are used for calibration refinement. More recently, coarse-to-fine Kinect V2 registration methods are proposed in [14, 15]. In particular, calibration objects, *e. g.*, marker (2D) and wand (1D), are applied in the coarse estimation stage. And Iterative Closest Point [16] and R-Nearest Neighbor [17] approaches are applied in the estimation refinement stage. Nevertheless, both of these two methods rely on specific calibration objects. How to directly make use of camera and scene constraints to register multi-Kinect V2 sensors has not been explored yet.

To solve the Kinect V2 registration problem in the large-displacement environment, a novel coarse-to-fine calibration method using camera and scene constraints is proposed in this paper. To be precise, an off-the-shelf feature detector is utilized to find constraints in the scene, homography and fundamental matrices are employed to construct constraints in the cameras. The coarse estimation is composed of feature point detection, coarse matching, match filtering, and least-squares fitting steps. The estimation refinement consists of an ICP-based point cloud registration algorithm. Experiments are conducted on the movable multi-camera rig with a large displacement between Kinect V2 cameras as introduced above. Experimental results show the validity of the proposed coarse-to-fine registration method using camera and scene constraints in the large-displacement environment.

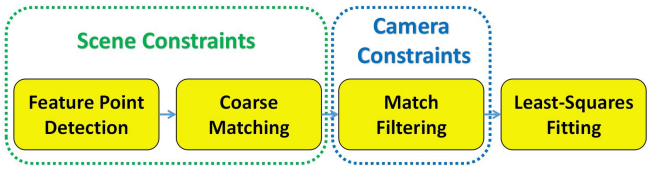


Fig. 2. Flow chart of the coarse estimation step with camera and scene constraints.

2. METHODOLOGY

In this section, coarse-to-fine registration methods, which consist of coarse estimation and estimation refinement steps, are presented after a brief preliminary description to introduce the problem.

2.1. Preliminary

Since there are two Kinect V2 cameras on the multi-camera rig, one Kinect V2 is denoted as C_A, the other one is denoted as C_B for convenience. The capture output of a Kinect V2 is a pair of registered depth and color images. The registered color images are shown in Fig. 3. More details are explained in section 3.1. Therefore, for each Kinect V2 camera, there are two basic coordinate systems. One is camera 3D space, the other is camera image space. The Kinect V2 registration problem can be defined as solving the rigid transformation from C_A 3D space to C_B 3D space, denoted as $\mathbf{R}$ and $\boldsymbol{t}$. The rigid transformation result of the coarse estimation step is denoted as $\mathbf{R}_1$ and $\boldsymbol{t}_1$. The incremental result of the estimation refinement step is denoted as $\mathbf{R}_2$ and $\boldsymbol{t}_2$. Suppose $\boldsymbol{u}_i^a = \begin{bmatrix} u_i^a & v_i^a & 1 \end{bmatrix}^{\mathrm{T}}$ is a point in C_A image space, the corresponding point $\boldsymbol{x}_i^a = \begin{bmatrix} x_i^a & y_i^a & z_i^a & 1 \end{bmatrix}^{\mathrm{T}}$ in the 3D space of C_A can be calculated using the intrinsic camera matrix $\mathbf{K}^a$. Suppose $\boldsymbol{x}_i^a$ and $\boldsymbol{x}_j^b$ correspond to the same point $\boldsymbol{x}_k$ in world 3D space, a good coarse-to-fine calibration should meet this condition:

$$\boldsymbol{x}_j^b = \begin{bmatrix} \mathbf{R}_2 & \boldsymbol{t}_2 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \tag{1}$$

where:

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_2\mathbf{R}_1 \\ \boldsymbol{t} &= \mathbf{R}_2\boldsymbol{t}_1 + \boldsymbol{t}_2 \end{aligned} \tag{2}$$

Therefore, (2) can be applied for recovering the final rigid transformation using the results of the two stages of coarse-to-fine registration methods.

2.2. Coarse Estimation

Two categories of coarse estimation approaches are introduced in this section. One is based on calibration objects. The other is based on camera and scene constraints.

2.2.1. With Calibration Objects

The calibration object is a tool of making some artificial geometric constraints for the assisted calibration in the coarse estimation stage. The checkerboard is one of the most common calibration tools in computer vision, which is also utilized here. After the corner detection step, n corresponding point pairs in C_A and C_B images spaces are detected. Each pair is expressed as $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ and $\boldsymbol{u}_i^a$ corresponds to a 3D point $\boldsymbol{x}_i^a$ as introduced in section 2.1. The coarse estimation problem can be solved by minimizing the following formula:

$$\begin{aligned} &\min_{\mathbf{R}_1,\boldsymbol{t}_1} \sum_{i=1}^{n} \|\boldsymbol{u}_i^b - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^a, \mathbf{K}^b, \mathbf{R}_1, \boldsymbol{t}_1)\|^2 \\ &\text{where } \hat{\boldsymbol{u}}(\boldsymbol{x}, \mathbf{K}, \mathbf{R}, \boldsymbol{t}) = \mathrm{proj}(\mathbf{K} \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \end{bmatrix} \boldsymbol{x}) \end{aligned} \tag{3}$$

Here, the Levenberg-Marquardt optimization algorithm is applied to solve this problem.

input : P_A - Point cloud of C_A after the rigid transformation of the coarse estimation stage;
P_B - Point cloud of C_B;
N - Number of point cloud registration iterations;
$\mathbf{R}^a, \mathbf{R}^b$ - 3×3 identity matrices;
$\boldsymbol{t}^a, \boldsymbol{t}^b$ - 3×1 zero vectors.
output: $\mathbf{R}^a, \mathbf{R}^b, \boldsymbol{t}^a, \boldsymbol{t}^b$.

for $n \leftarrow 1$ **to** N **do**
  $\mathbf{R}, \boldsymbol{t} \leftarrow ICP(P_A, P_B)$;
  for *each point* $\boldsymbol{x}_i^a$ *in* P_A **do**
    $\boldsymbol{x}_i^a \leftarrow \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a$;
  end
  $\mathbf{R}^a \leftarrow \mathbf{R}\mathbf{R}^a$;
  $\boldsymbol{t}^a \leftarrow \mathbf{R}\boldsymbol{t}^a + \boldsymbol{t}$;
  $\mathbf{R}, \boldsymbol{t} \leftarrow ICP(P_B, P_A)$;
  for *each point* $\boldsymbol{x}_i^b$ *in* P_B **do**
    $\boldsymbol{x}_i^b \leftarrow \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^b$;
  end
  $\mathbf{R}^b \leftarrow \mathbf{R}\mathbf{R}^b$;
  $\boldsymbol{t}^b \leftarrow \mathbf{R}\boldsymbol{t}^b + \boldsymbol{t}$;
end

Algorithm 1: Point cloud registration

2.2.2. With Camera and Scene Constraints

There are four basic steps in the coarse estimation stage using camera and scene constraints as illustrated in Fig. 2. Scene constraints are calculated in the first two steps. Camera constraints help reject outliers in the third step. Details of these four steps are explained as below.

Feature point detection: Interest point detection is a well studied research area in computer vision [18]. Speeded Up Robust Features (SURF) [19] is used here for the reason that it is a fast and robust feature descriptor which is suitable for the feature detection task of this work. Besides, there are some holes in the registered color images of Kinect V2 sensors, which result in the bad performance of some other interest point detectors. Comparisons of different point detection methods can be done in the Full High Definition (FHD) resolution images of the Kinect V2 RGB sensor, while the exact transformation relationship between RGB and ToF sensors in a Kinect V2 is unknown but studied in [20, 21, 22], which is beyond the research scope of this paper.

Coarse matching: The k-Nearest-Neighbors (KNN) [23] and the ratio test [24] are utilized to match the detected feature points from the image spaces of two cameras.

Match filtering: In this step, camera constraints are utilized to filter the outlier matches with the RANdom SAmple Consensus (RANSAC) [25] framework. Camera constraints include homography and fundamental matrices [26] described as:

$$\begin{aligned} \boldsymbol{u}_i^b &\sim \mathbf{H}\boldsymbol{u}_i^a \\ (\boldsymbol{u}_i^b)^{\mathrm{T}}\mathbf{F}\boldsymbol{u}_i^a &= 0 \end{aligned} \tag{4}$$

Here, $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ is a corresponding point pair in the camera image planes of C_A and C_B after the coarse matching process. The RANSAC algorithm is employed to calculate $\mathbf{H}$ and $\mathbf{F}$ robustly, which are then used to keep the inlier matches.

Least-squares fitting: The inlier matches $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ are transformed to $(\boldsymbol{x}_i^a, \boldsymbol{x}_i^b)$ in the camera 3D spaces of C_A and C_B using

Table I. RMSE of the coarse-to-fine registration methods.

Coarse Estimation	RMSE (mm)	Estimation Refinement	RMSE (mm)
Checkerboard-based	**78.33**	ICP-based point cloud registration	84.11
Homography matrix-based	302.05	ICP-based point cloud registration	44.82
Fundamental matrix-based	295.58	ICP-based point cloud registration	**34.34**

$\mathbf{K}^a, \mathbf{K}^b$ and the depth information from the registered depth image. The least-squares fitting algorithm [27] is the process of minimizing the distance between two 3D point sets using the Singular Value Decomposition (SVD) method:

$$\min_{\mathbf{R}_1, \boldsymbol{t}_1} \sum_{i=1}^{n} \| \boldsymbol{x}_i^b - \begin{bmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \|^2 \tag{5}$$

2.3. Estimation Refinement

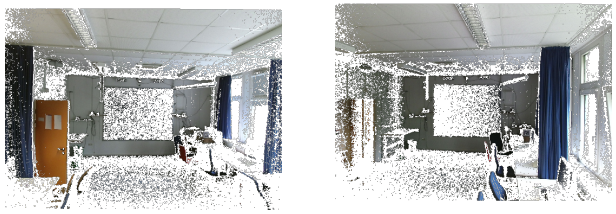
The Iterative Closest Point (ICP) algorithm is an effective method for the registration of two similar point clouds without explicitly given correspondences [28]. The reason why ICP is chosen here is that it has similar performance in this experimental setup compared with other methods, *e. g.*, R-Nearest Neighbor [17], 3D feature detection and matching [29], which is also stated in [15]. A common point cloud registration algorithm for point clouds of two cameras is illustrated in Algorithm 1. Note that P_A is a transformed point cloud of C_A after the coarse estimation stage. Suppose $\boldsymbol{x}_i^a$ is a point in P_A, which corresponds to a point $\boldsymbol{x}_j^b$ in P_B. Both of them also correspond to the same point $\boldsymbol{x}_k$ in world 3D space. After using the point cloud registration algorithm, the following formula should hold ideally:

$$\boldsymbol{x}_k = \begin{bmatrix} \mathbf{R}^a & \boldsymbol{t}^a \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a = \begin{bmatrix} \mathbf{R}^b & \boldsymbol{t}^b \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_j^b \tag{6}$$

Therefore, the rigid transformation of estimation refinement step is:

$$\begin{aligned} \mathbf{R}_2 &= (\mathbf{R}^b)^{\mathrm{T}} \mathbf{R}^a \\ \boldsymbol{t}_2 &= (\mathbf{R}^b)^{\mathrm{T}} (\boldsymbol{t}^a - \boldsymbol{t}^b) \end{aligned} \tag{7}$$

(a) C_A view of the checkerboard (b) C_B view of the checkerboard



(c) C_A view of the scene (d) C_B view of the scene

Fig. 3. Registered color images for experiments.

3. EXPERIMENTS

3.1. Experimental Settings

Experiments are implemented on the movable multi-camera rig as introduced in section 1. Two Kinect V2 cameras C_A and C_B are connected to only one control computer using the open-source Kinect V2 driver, *libfreenect2*[1]. The capture process of both cameras is synchronized internally with signal and time stamp technologies. Using this Kinect V2 driver, the intrinsic camera matrices $\mathbf{K}^a$ and $\mathbf{K}^b$ can be accessed. Besides, the captured views of Kinect V2 sensors are pairs of registered color and depth images with lens distortion corrected. An example output registered color image is shown in Fig. 3 (a). Note that each color pixel in this image has a corresponding depth value in the paired registered depth image, and pixels with color information missing mean that depth information for these pixels can not be accessed from the sensors. The resolutions of both registered color and depth images are 512×424 pixels.

Checkerboard-captured data: A checkerboard is put in front of the multi-camera rig at a distance of around 2.8 m. This checkerboard has 28 (4×7) inner corners and the size of each square in it is 124×124 mm. Registered color images for the views of this checkerboard in C_A and C_B are illustrated in Fig. 3 (a)(b). This data is used for the coarse estimation step with a calibration object as described in section 2.2.1 and the evaluation metric as described below.

Scene-captured data: A natural scene of a conference room of the size of $5.5 \times 3.0 \times 7.8$ m ($w \times h \times d$) is captured without artificial calibration objects in it. Registered color images for the views of this scene in C_A and C_B are illustrated in Fig. 3 (c)(d). This data is used for the coarse estimation with camera and scene constraints from section 2.2.2 and the estimation refinement step from section 2.3.

Coarse estimation details: Automatic corner detection and SURF detection are implemented with OpenCV. Parameter r for ratio test is set to 0.6. The threshold ε in RANSAC is set to 1.0 pixels.

Estimation refinement details: The number of point cloud registration iterations N is set to 10. In each point cloud registration iteration, the ICP function has 5 iterations.

Evaluation metric: The Root-Mean-Square Error (RMSE) is adopted to evaluate the effects of coarse-to-fine registration methods with the checkerboard-captured data. Using the same corresponding point pair definition $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ as described in section 2.2.1, the number of corresponding point pairs n is equal to 28. The RMSE is defined as:

$$\sqrt{\frac{1}{n} \sum_{i=1}^{n} \| \boldsymbol{x}_i^b - \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \|^2} \tag{8}$$

Comparison experiments are conducted on an Intel Core i3 − 4030U laptop with 16 GB RAM and no GPU acceleration, using the captured datasets from the control computer. Both source code and datasets are going to be released on our website[2].

3.2. Results and Analysis

Quantitative evaluation results of different coarse-to-fine registration methods are exhibited in Table I. In the coarse estimation phase,

[1] http://dx.doi.org/10.5281/zenodo.50641

[2] https://ygaokiel.github.io

(a) Checkerboard-based coarse estimation with estimation refinement

(b)

(c) Fundamental matrix-based coarse estimation with estimation refinement

(d)

Fig. 4. Results of registered point clouds with coarse-to-fine registration methods. Red circle areas of (a) and (c) are amplified in (b) and (d).

the checkerboard-based method achieves much more precise results than the homography and fundamental matrices-based methods, which shows that the calibration object is an effective assistance tool to improve the initial calibration. Then, in the estimation refinement phase, the precision of the checkerboard-based method decreases a little bit, while the precision of coarse estimation methods using camera and scene constraints improves dramatically. Specifically, fundamental matrix-based coarse estimation plus estimation refinement has the best performance among these three coarse-to-fine registration methods. The reason for this may be that the coarse estimation with camera and scene constraints gives a globally optimal start point for the ICP-based point cloud registration method, while the coarse estimation using the calibration object offers a locally optimal start point, which leads to its failure in this large-displacement setup. Also, ICP remedies wrong matches that might still remain after the filtering in the coarse estimation step.

Qualitative evaluation results are also presented as shown in Fig. 4 using the scene-captured data. Red circles in Fig. 4 (a)(c) indicate the same white box on the table. The result of checkerboard-based coarse estimation with estimation refinement in Fig. 4 (a) has obvious non-overlapping parts in the red circle area. However, in Fig. 4 (c), the two point clouds coincide very well in the location marked with the red circle, which indicates the effectiveness of coarse estimation with camera and scene constraints again.

4. CONCLUSION

In this paper, camera and scene constraints are exploited inside a coarse-to-fine framework to solve the Kinect V2 registration problem in the large-displacement environment. The proposed Kinect V2 registration method uses homography and fundamental matrix estimations from 2D correspondences found with a SURF detector to estimate the camera pose. The fundamental matrix-based coarse-to-fine registration method outperforms the checkerboard-based coarse-to-fine registration method on a multi-camera rig with a large displacement between two Kinect V2 sensors, which proves the effectiveness of the proposed Kinect V2 registration method for large-displacement environments. How to exploit the camera and scene constraints in the FHD-resolution RGB sensors of Kinect V2 devices to solve the Kinect V2 registration problem in the large-displacement environment will be our next research goal.

5. ACKNOWLEDGMENTS

The work in this paper was funded from the European Union′s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, Intel-VCI-CAU, the German Research Foundation (DFG) No. K02044/8-1 and the Fraunhofer and Max Planck cooperation program within the German pact for research and innovation (PFI).

6. REFERENCES

[1] Andrea Corti, Silvio Giancola, Giacomo Mainetti, and Remo Sala, "A metrological characterization of the Kinect v2 Time-of-Flight camera," *Robotics and Autonomous Systems (RAS)*, vol. 75, pp. 584–594, 2016.

[2] Oliver Wasenmüller and Didier Stricker, "Comparison of Kinect v1 and v2 depth images in terms of accuracy and precision," in *Asian Conference on Computer Vision Workshops (ACCVW)*, 2016.

[3] Marek Kraft, Michał Nowicki, Adam Schmidt, Michał Fularz, and Piotr Skrzypczyński, "Toward evaluation of visual navigation algorithms on RGB-D data from the first- and second-generation Kinect," *Machine Vision and Applications (MVA)*, pp. 1–14, 2016.

[4] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb, "Kinect range sensing: Structured-light versus Time-of-Flight Kinect," *Computer Vision and Image Understanding (CVIU)*, vol. 139, pp. 1–20, 2015.

[5] S Zennaro, M Munaro, S Milani, P Zanuttigh, A Bernardi, S Ghidoni, and E Menegatti, "Performance evaluation of the 1st and 2nd generation Kinect for multimedia applications," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[6] Matteo Munaro, Filippo Basso, and Emanuele Menegatti, "Openptrack: Open source multi-camera calibration and people tracking for RGB-D camera networks," *Robotics and Autonomous Systems (RAS)*, vol. 75, pp. 525–538, 2016.

[7] Can Wang, Hong Liu, and Yuan Gao, "Scene-adaptive hierarchical data association for multiple objects tracking," *IEEE Signal Processing Letters (SPL)*, vol. 21, no. 6, pp. 697–701, 2014.

[8] Andrea Canessa, Manuela Chessa, Agostino Gibaldi, Silvio P Sabatini, and Fabio Solari, "Calibrated depth and color cameras for accurate 3D interaction in a stereoscopic augmented reality environment," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 227–237, 2014.

[9] Licong Zhang, Jürgen Sturm, Daniel Cremers, and Dongheui Lee, "Real-time human motion tracking using multiple depth cameras," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 2389–2395.

[10] Sandro Esquivel, Yuan Gao, Tim Michels, Luca Palmieri, and Reinhard Koch, "Synchronized data capture and calibration of a large-field-of-view moving multi-camera light field rig," in *3DTV-CON Workshops*, 2016.

[11] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 22, no. 11, pp. 1330–1334, 2000.

[12] Petar Palasek, Heng Yang, Zongyi Xu, Navid Hajimirza, E-broul Izquierdo, and Ioannis Patras, "A flexible calibration method of multiple Kinects for 3D human reconstruction," in *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2015, pp. 1–4.

[13] Stephan Beck and Bernd Froehlich, "Volumetric calibration and registration of multiple RGBD-sensors into a joint coordinate system," in *IEEE Symposium on 3D User Interfaces (3DUI)*, 2015, pp. 89–96.

[14] Marek Kowalski, Jacek Naruniec, and Michal Daniluk, "LiveScan3D: A fast and inexpensive 3D data acquisition system for multiple Kinect v2 sensors," in *IEEE International Conference on 3D Vision (3DV)*, 2015, pp. 318–325.

[15] Diana-Margarita Córdova-Esparza, Juan R Terven, Hugo Jiménez-Hernández, and Ana-Marcela Herrera-Navarro, "A multiple camera calibration and point cloud fusion tool for Kinect v2," *Science of Computer Programming (SCP)*, 2016.

[16] Paul J Besl and Neil D McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 14, no. 2, pp. 239–256, 1992.

[17] Alexandr Andoni, *Nearest neighbor search: the old, the new, and the impossible*, Ph.D. thesis, Massachusetts Institute of Technology, 2009.

[18] Steffen Gauglitz, Tobias Höllerer, and Matthew Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International Journal of Computer Vision (IJCV)*, vol. 94, no. 3, pp. 335–360, 2011.

[19] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "SURF: Speeded up robust features," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 404–417.

[20] Yuan Gao, Matthias Ziegler, Frederik Zilly, Sandro Esquivel, and Reinhard Koch, "A linear method for recovering the depth of Ultra HD cameras using a Kinect v2 sensor," in *IAPR International Conference on Machine Vision Applications (MVA)*, 2017, pp. 464–467.

[21] Wanbin Song, Anh Vu Le, Seokmin Yun, Seung-Won Jung, and Chee Sun Won, "Depth completion for Kinect v2 sensor," *Multimedia Tools and Applications (MTA)*, pp. 1–24, 2016.

[22] Oliver Wasenm, Marcel Meyer, and Didier Stricker, "Corbs: Comprehensive RGB-D benchmark for slam using Kinect v2," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–7.

[23] Naomi S Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[24] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[25] Martin A Fischler and Robert C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] Richard Hartley and Andrew Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[27] K. Somani Arun, Thomas S. Huang, and Steven D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. PAMI-9, no. 5, pp. 698–700, 1987.

[28] François Pomerleau, Francis Colas, Roland Siegwart, and Stéphane Magnenat, "Comparing ICP variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.

[29] Tal Darom and Yosi Keller, "Scale-invariant features for 3-D mesh models," *IEEE Transactions on Image Processing (TIP)*, vol. 21, no. 5, pp. 2758–2769, 2012.