

Marker-Based Extrinsic Calibration Method for Accurate Multi-Camera 3D Reconstruction

Nahuel Garcia-D'Urso

`nahuel.garcia@ua.es`

University of Alicante

Bernabe Sanchez-Sos

University of Alicante

Jorge Azorin-Lopez

University of Alicante

Andres Fuster-Guillo

University of Alicante

Antonio Macia-Lillo

University of Alicante

Higinio Mora-Mora

University of Alicante

Research Article

Keywords: Multi-camera calibration, RGB-D, 3D Reconstruction, Marker-based calibration

Posted Date: May 27th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-9666210/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Marker-Based Extrinsic Calibration Method for Accurate Multi-Camera 3D Reconstruction

Nahuel Garcia-D'Urso^{1*†}, Bernabe Sanchez-Sos^{1†},
Jorge Azorin-Lopez^{1†}, Andres Fuster-Guillo^{1†},
Antonio Macia-Lillo^{1†}, Higinio Mora-Mora^{1†}

^{1*}Department of Computer Science and Technology (DTIC), University of Alicante, San Vicente del Raspeig, Alicante, 03690, Spain.

*Corresponding author(s). E-mail(s): nahuel.garcia@ua.es;
Contributing authors: bernabe.sanchez@ua.es; jazorin@ua.es;
fuster@ua.es; a.macia@ua.es; hmora@ua.es;

[†]These authors contributed equally to this work.

Abstract

Accurate 3D reconstruction using multi-camera RGB-D systems critically depends on precise extrinsic calibration to achieve proper alignment between captured views. In this paper, we introduce an iterative extrinsic calibration method that leverages the geometric constraints provided by a three-dimensional marker to significantly improve calibration accuracy. Our proposed approach systematically segments and refines marker planes through clustering, regression analysis, and iterative reassignment techniques, ensuring robust geometric correspondence across camera views. We validate our method comprehensively in both controlled environments and practical real-world settings within the Tech4Diet project, aimed at modeling the physical progression of patients undergoing nutritional treatments. Experimental results demonstrate substantial reductions in alignment errors, facilitating accurate and reliable 3D reconstructions.

Keywords: Multi-camera calibration, RGB-D, 3D Reconstruction, Marker-based calibration

1 Introduction

In recent years, the demand for high-precision 3D human models has significantly grown across diverse industries, including fashion and garment design, anthropometric measurement extraction, virtual and augmented reality, and gaming. This increasing interest has propelled advancements in multi-sensor technologies, offering enhanced precision and versatility in capturing detailed 3D data.

Among various technologies employed for generating accurate 3D models, RGB-D cameras have emerged as highly effective solutions due to their ability to combine standard RGB imaging with depth sensing. These cameras offer cost-effective, robust, and detailed 3D representations, making them popular in numerous applications. Different RGB-D camera technologies such as structured light, time-of-flight, and stereoscopic systems have distinct advantages tailored to specific application requirements.

Despite their widespread adoption, RGB-D cameras still encounter substantial challenges, primarily related to calibration, which significantly impacts the quality of resulting reconstructions. Accurate calibration is crucial in multi-camera systems and involves determining both intrinsic parameters for individual sensors and extrinsic parameters for camera sets. Calibration errors can result in misaligned depth and color data, severely compromising the integrity and precision of the reconstructed models. Numerous calibration techniques, including iterative and marker-based approaches, have been developed to address these issues, yet perfect integration between multiple sensors remains challenging [1–3].

Beyond calibration, environmental conditions such as lighting variations, occlusions, and sensor noise significantly influence RGB-D camera performance. Each sensor type exhibits particular vulnerabilities; structured light systems struggle in poor or overly intense lighting, whereas time-of-flight sensors face accuracy reductions at longer distances or in dynamic scenarios. These limitations necessitate sophisticated algorithms to enhance robustness and adaptability [2, 4, 5].

Additionally, multi-camera arrangements inherently introduce complexity, demanding precise synchronization and alignment to accurately reconstruct 3D scenes. Although significant advancements in calibration and multi-view reconstruction have been made, persistent issues such as depth measurement errors and the requirement for extensive post-processing underscore the ongoing need for research that further refines RGB-D system capabilities [3, 6].

Within this context, the Tech4Diet project focuses on generating 4D models (3D plus temporal dimension) of patients undergoing nutritional treatments. The objective is to accurately capture patient body changes across multiple sessions, presenting their physical progression through augmented reality visualization. This visualization serves as a motivational tool, increasing patient engagement, enhancing treatment adherence, reducing dropout rates, and ultimately improving weight-loss outcomes.

In this paper, we propose an iterative, marker-based calibration methodology specifically designed for multi-camera RGB-D systems. Our approach employs a three-dimensional cube-shaped marker that offers key advantages over traditional planar or spherical calibration objects. Thanks to its geometric design, the marker ensures that multiple planes are simultaneously visible from almost any viewpoint, significantly enhancing robustness and simplifying capture requirements. Unlike spherical-based

methods, which typically fit a single model and are highly sensitive to ellipse fitting accuracy, our method independently fits three planar models—one per visible face—followed by the enforcement of strict orthogonality constraints between them. This multi-model fitting strategy improves both the local and global calibration robustness, enabling accurate alignment even under suboptimal viewing conditions or partial occlusions. As a result, the proposed method offers a versatile and highly reliable solution for extrinsic calibration across diverse multi-camera configurations. To evaluate our method’s effectiveness, we conducted three primary experiments examining different configurations of marker height and position. Furthermore, we optimized system hyperparameters to enhance calibration accuracy, demonstrating the applicability of our approach through the successful reconstruction of human body 3D models. The implementation of our proposed calibration method and dataset is publicly available at: [Github](#).

The remainder of the paper is structured as follows: Section 2 reviews the relevant literature, Section 3 outlines the materials and methods, including camera setup details, marker design specifics, and the reconstruction methodology. Section 4 describes the experimental setups and discusses the obtained results. Finally, Section 5 provides concluding remarks and highlights potential future research directions.

2 Related Work

Recent advancements in RGB-D camera technology have greatly enhanced the capabilities of multi-camera systems, enabling a diverse range of applications such as autonomous driving, indoor navigation, and 3D reconstruction. A crucial step in achieving precise multi-view fusion and accurate 3D reconstructions is extrinsic calibration, which defines the spatial relationships among the cameras. Traditionally, extrinsic calibration has relied on planar targets such as checkerboards or calibration boards. While effective in controlled environments, these methods often prove inadequate in complex multi-camera configurations. Numerous methods have been proposed to tackle this challenge. Gao et al. [7] presented a methodology utilizing sparse point clouds derived from a 3D object with a texture-rich surface to simultaneously estimate intrinsic and extrinsic camera parameters. Their approach reduces the calibration complexity by employing graph-based optimization techniques. Dai et al. [8] further addressed the calibration of non-overlapping multi-camera systems, a particularly challenging scenario, by generating independent sparse 3D maps and subsequently determining optimal feature correspondences to estimate extrinsic parameters. Shen et al. [9] built a non-planar calibration object with sphere and formed a visual sensor network with multiple cameras to calibrate multiple cameras. However, the precision of the method based on sphere is greatly influenced by the precision of ellipse fitting.

Mehmandar et al. [10] presented a neural network-based recalibration framework for real-time adjustment of infrared multi-camera systems. Their approach dynamically adjusts camera poses using a differentiable projection model, thereby enhancing robustness against environmental perturbations. Concurrently, Dexheimer et al. [11] proposed an information-theoretic approach focused on online extrinsic calibration.

Their methodology minimizes entropy and selects informative keyframes to achieve robust and computationally efficient calibration in dynamic settings.

Addressing dynamic and complex environments, recent research has integrated calibration processes within simultaneous localization and mapping (SLAM) frameworks. Dynamic object handling and robust extrinsic calibration are emphasized to cope with real-world complexities. This integrated calibration and SLAM approach facilitates robust extrinsic parameter estimation even in environments with significant scene dynamics and occlusions [12].

Multi-sensor fusion involving cameras and LiDAR sensors has also seen notable advancements. Grammatikopoulos et al. [13] proposed a straightforward yet robust spatiotemporal calibration method for camera-LiDAR systems, utilizing a simple calibration target to effectively align spatial and temporal measurements. This method greatly improves the accuracy and reliability of sensor data fusion in mobile mapping scenarios.

Moreover, recent methods specifically tailored for RGB-D camera setups have emerged. Shin et al. [14] proposed a targetless calibration method called "PeLiCal," leveraging line features to robustly calibrate systems with limited camera overlap. Curto et al. [6] described a multi-camera RGB-D setup optimized for reconstructing deformable objects, using bright-spot trajectories for extrinsic calibration. Additionally, He et al. [15] introduced an automatic extrinsic calibration approach for infrastructure RGB-D networks with small fields of view, utilizing a moving checkerboard and pose graph optimization to enhance robustness.

In summary, while most recent methods for extrinsic calibration in multi-camera systems impose strict requirements, such as reliance on specific RGB-D sensor types, rigid camera setups, or highly controlled marker designs, our proposed approach offers greater flexibility. Although it employs a dedicated three-dimensional cube-shaped marker to exploit geometric constraints, it remains sensor-agnostic and adaptable to arbitrary multi-camera configurations. The marker's structure ensures that multiple faces are visible from almost any viewpoint, overcoming visibility limitations typically associated with planar or spherical targets. Moreover, rather than fitting a single model, as in sphere-based methods that are sensitive to ellipse fitting errors, our method independently fits three planes—one for each visible face—and enforces orthogonality constraints between them. This multi-model fitting strategy significantly enhances calibration robustness and accuracy. The only prerequisite for our method is ensuring overlapping views between cameras, enabling robust and scalable calibration in a wide range of practical deployment scenarios.

3 Materials and Methods

3.1 Notations and Setting

The multi-camera system consists of m cameras organized in a matrix, named as $C = \{C_1^{1,1}, C_2^{1,2}, \dots, C_m^{i,j}\}$, where i and j indicate respectively the row and column in the camera system. Each camera $C_m^{i,j}$ captures a 3D point cloud $P_m^{i,j}$ and an RGB image $I_m^{i,j}$. The point cloud consists of a set of points $\{p_1, p_2, \dots, p_n\}$ in Euclidean

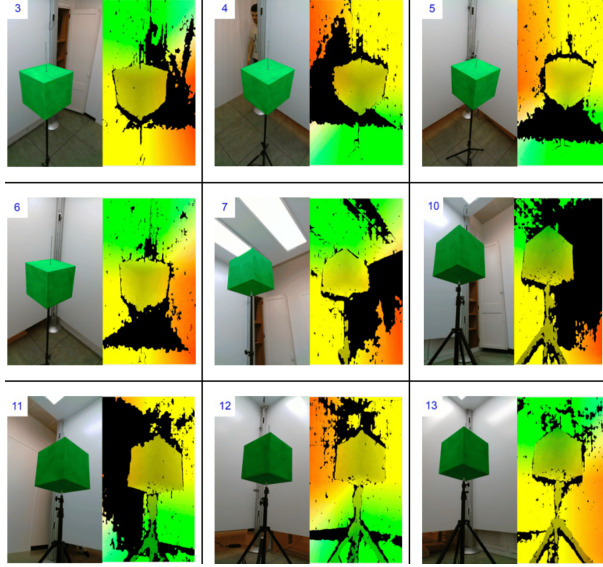


Fig. 1: Example of RGB and depth images of the cube used for calibration.

space, each point being $p_n = [x_n, y_n, z_n]$. A plane π is defined as:

$$\pi : \mathbf{n}^\top \mathbf{x} + d = 0, \quad (1)$$

where $\mathbf{n} \in \mathbb{R}^3$ is the unit normal vector, $d \in \mathbb{R}$ is the distance to the origin, and $\mathbf{x} \in \mathbb{R}^3$ represents any point on the plane. For calibration, we use a cube consisting of six planar surfaces $\{\pi_1, \pi_2, \dots, \pi_6\}$, which satisfies the following geometric constraints:

- **Orthogonality:** The normal vectors of the planes satisfy $\mathbf{n}_i^\top \mathbf{n}_j = 0, \forall i \neq j$.
- **Constant edge length:** All edges of the cube have a known length.
- **Visibility constraint:** A camera observes at most three planes of the cube simultaneously.

3.2 Multi-Camera System Calibration

In this section, we present the proposed methodology for obtaining the transformation matrices (extrinsic calibration) of an RGB-D camera system. The method is divided into five main stages. Firstly, in Section 3.3, color and depth images are processed. Then, in Section 3.4, the marker planes are extracted, following an adaptation of the approach presented in [16, 17]. Finally, in Section 3.5, the calibration matrices that align the different cameras are estimated. The marker used is a cube, as its known geometry allows for the exploitation of robust geometric constraints, such as orthogonality between planes and partial visibility from each camera. Each camera must simultaneously visualize three faces of the cube for the method to function correctly, as shown in Figure 1.

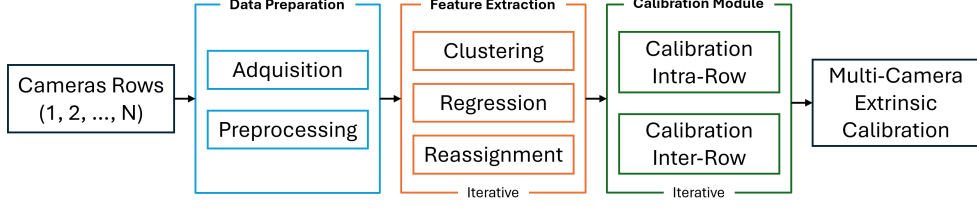


Fig. 2: Overview of the proposed calibration pipeline. The method is structured into three main modules: Data Preparation, Feature Extraction, and Calibration. Each module includes specific processing stages, starting from RGB-D image acquisition and preprocessing, followed by iterative geometric extraction of cube faces, and ending with intra and inter-row calibration steps. The final output is the extrinsic calibration of the multi-camera system.

3.3 Data Preparation

This stage aims to extract the region of interest from RGB and depth images captured by each camera. It is assumed that all cameras have been previously calibrated intrinsically. Before initiating this process, it is necessary to capture images of the calibration cube at various heights and positions, ensuring at least one capture exists in which cameras from different rows simultaneously observe the cube. Given that the calibration cube is green, the RGB images are first converted to the HSV color space to facilitate effective color-based segmentation. A binary mask is subsequently created by applying predefined thresholds to the hue, saturation, and depth channels. Using region detection, the largest connected region, corresponding to the marker, is identified and segmented in both RGB and depth images. From this segmentation, a filtered 3D point cloud is obtained, containing only the cube region to be used in the subsequent stage.

3.4 Feature Extraction

This stage focuses on extracting geometric information from the segmented point cloud $P_m^{i,j}$ to identify the visible faces of the cube and estimate their corresponding planes. The process comprises three main steps clustering, regression, and reassignment which are applied iteratively.

The clustering stage involves identifying and grouping 3D points from the segmented point cloud $P_m^{i,j}$ into distinct flat regions that correspond to the visible faces of the cube. The objective of the clustering stage is to segment the point cloud into N groups, where N corresponds to the number of visible planes of the cube (three per camera view). This process uses both the 3D coordinates of the points $\mathbf{p} = [x, y, z]$ and their normal vectors $\mathbf{n} = [n_x, n_y, n_z]$. Each point's normal is computed by averaging the normals of its neighboring points.

Points and their normals are grouped using the K-means algorithm, where each resulting cluster represents a plane π corresponding to one of the visible faces of the

cube. Clusters with insufficient point density are discarded to improve robustness. Additionally, the angular consistency of each group is verified by comparing the angles between individual normal vectors and the median group’s normal vector. Points with large angular deviations are excluded from the cluster.

To ensure that each group represents a single plane π , groups with similar normal vectors or those that are spatially close are merged. This merging process uses the orthogonality constraint. Since the object to be reconstructed is known (a cube), it is understood that the angle between different planes must always be 90 degrees. Finally, for each obtained cluster, its centroid $\mathbf{c}$ and normal vector $\mathbf{n}$ are calculated.

The regression stage refines the planes obtained during clustering by fitting each group of points to a mathematical representation of a cube (identical to the real one). The aim of the regression is to estimate the parameters $\mathbf{n}$ and d of a plane that minimize the error, that is, the distance between the points of the group and the plane of a cube model.

For a group of points $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ associated with a plane π , the regression process begins by calculating the centroid of the points:

$$\mathbf{C} = \frac{1}{N} \sum_{i=1}^N \mathbf{p}_i. \quad (2)$$

Then, the normal vector $\mathbf{n}$ is calculated using principal component analysis (PCA). The ultimate goal of this step is to verify that the obtained clusters meet the necessary orthogonality constraints to create a cube. That is, there is a 90-degree angle between them, thus enabling the generation of a cube.

The reassignment stage seeks to validate the belonging of a point to a plane. For each point $\mathbf{p}_i$ in the segmented point cloud $Pm^{i,j}$, its distance to all candidate planes $\{\pi_1, \pi_2, \pi_3\}$ is calculated using the plane equation 1. The point is reassigned to the plane π_k that minimizes this distance, ensuring that each point is associated with the closest plane. Then, the similarity of the point’s normal with the normal of the assigned plane is checked. To calculate the point’s normal, the average between its normal and those of adjacent points is computed.

In the next step, the angular consistency between the point’s normal vector $\mathbf{n}_{\mathbf{p}_i}$ and the plane’s normal vector $\mathbf{n}_{\pi_i}$ is evaluated. Points with angular deviations greater than a predefined threshold are excluded. Only points that meet the angular constraint are kept in the reassigned groups. This primarily occurs on the edges, where a point may be close to a plane but that plane has a different normal.

The reassignment process is carried out iteratively to refine the association between points and planes. After each iteration, the n and d parameters of the planes are recalculated using the updated groups, and the reassignment is repeated. The Clustering, Regression, and Reassignment stages are repeated iteratively until all points belong to their corresponding plane. This iterative refinement continues until all points belong to their respective plane. An example of the iterative process of obtaining the planes of the cube can be observed in the figures 3 and 4.

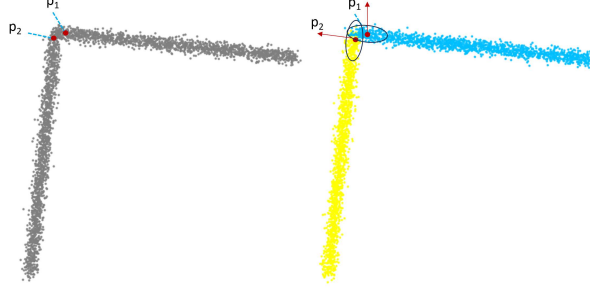


Fig. 3: Example of two points erroneously assigned to the same plane.

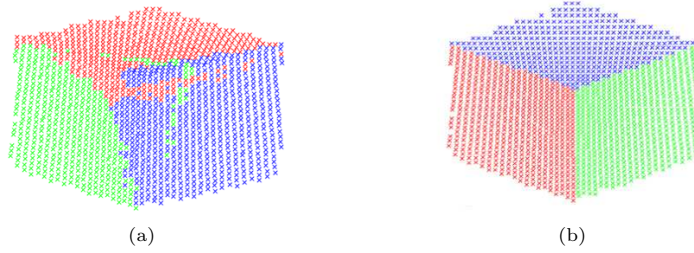


Fig. 4: Comparison of cluster assignments in different phases of the algorithm. (a) Cube model with points assigned in wrong clusters at the beginning of the algorithm. (b) The same cube after iterating multiple times over the Cluster, Regression, and Reassignment phases.

3.5 Calibration

The calibration step computes the transformation matrices that align the multi-camera system $C = \{C_1^{1,1}, C_2^{1,2}, \dots, C_m^{i,j}\}$ by estimating the relative positions and orientations of all cameras within a common reference frame. Each transformation matrix $\mathbf{T}_m^{i,j} \in SE(3)$ is a rigid 4×4 transformation composed of a rotation matrix $\mathbf{R} \in SO(3)$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$, which together describe the pose of the camera in space.

To compute these transformations, the calibration algorithm utilizes geometric features extracted from a known marker. Specifically, it focuses on the centroids and normals of the three visible faces of a cube in each camera's view. To enhance robustness against outliers and noise, a traditional RANSAC (Random Sample Consensus) method is employed to eliminate inconsistent observations. Following this, a Procrustes analysis is used to estimate the transformation that best aligns the observed planes from each camera to a chosen reference view. The alignment process is conducted jointly on the centroids and normals to ensure both positional and angular consistency.

The algorithm structures the camera system as a graph, where each node corresponds to a camera and edges represent pairs of cameras with simultaneous

observations of the calibration marker. Calibration is then performed in two sequential stages.

1. **Intra-row calibration:** For each row of cameras, a reference camera is selected, and its transformation matrix is fixed to the identity, $T = I$. The other cameras in the same row are then calibrated relative to this reference camera by estimating the rigid transformations needed to align their observed planes.
2. **Inter-row calibration:** After each row has been calibrated internally, the rows are connected using overlapping views. For example, cameras in row 2 that share views with cameras in row 1 are used to calculate the relative transformations between the rows. The same procedure is repeated between all rows. This results in a complete set of transformation matrices for all the cameras in the system.

This graph-based strategy allows for a scalable and structured calibration, even in systems with limited overlap between some camera pairs. During each alignment step, an error metric is computed to assess calibration quality. This metric combines two components: the average Euclidean distance between corresponding centroids and the mean angular deviation between their normal vectors. The total alignment error is defined as a weighted sum of both terms:

$$\text{Error} = \alpha \cdot \text{Distance Error} + \beta \cdot \text{Angular Error}, \quad (3)$$

where α and β are weighting factors that control the relative importance of spatial and directional accuracy.

Once the iterative process converges, each camera will have an associated transformation matrix $\mathbf{T}_m^{i,j}$. This matrix allows the point cloud from each camera to be expressed in a unified coordinate system defined by the reference camera. This approach ensures spatial consistency across the entire camera system, enabling accurate and coherent 3D reconstruction.

4 Experimentation

4.1 Experimental Setup

Two experiments were conducted to quantitatively validate the proposed method. The first experiment assesses the number of captures necessary to accurately calculate the marker planes. The objective is to determine the optimal number of captures and cube positions required for precise calibration. The second experiment focuses on optimizing the hyperparameters of our method. Specifically, we aim to identify the best parameter settings for reconstructing an object within the capture zone.

To evaluate the quality of the calibration matrix obtained with our method, we use various metrics. The size difference between the reconstructed and ground truth objects is computed. Additionally, we calculate the angle, in degrees, of the reconstructed cube’s planes. This is done for each row of cameras and for the entire system as a whole.

Furthermore, we use Wasserstein and Hausdorff distances to evaluate the accuracy of our system in reconstructing human body models.

The Hausdorff distance between two sets of points A and B is defined as:

$$d_H(A, B) = \max \left\{ \sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(a, b) \right\}, \quad (4)$$

where $d(a, b)$ represents the distance between the points $a \in A$ and $b \in B$. This metric measures the maximum distance from a point in one set to the closest point in the other set.

The Wasserstein distance quantifies the minimum effort required to transform one distribution P into another Q . In the context of 3D reconstruction, it evaluates the structural similarity between the reconstructed and reference models.

Additionally, we present qualitative results. For this, we perform the reconstruction of a human body using different calibrations obtained during experimentation. This human body belongs to one of the patients examined in the Tech4Diet project.

To determine the optimal camera configuration, a 3D simulation was carried out using Blender, figure 5. This setup aimed to maximize coverage of the scanned volume, minimize occlusions, and ensure sufficient overlap between adjacent views for accurate reconstruction. The final synthetic configuration consists of 12 RGB cameras. The Intel RealSense camera was selected due to its low cost, compact design, and wide field of view.

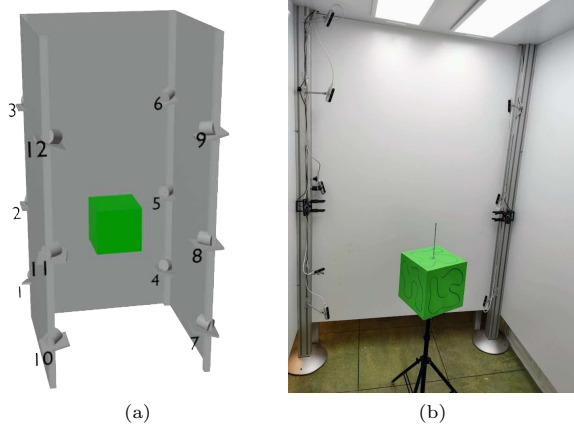


Fig. 5: On the left, the 3D simulation of the booth using Blender. On the right, the designed booth prototype.

4.2 Quantitative Results

In Table 1, the results obtained by reconstructing the cube in different scenarios can be observed. The first column shows the number of captures made for each position of the cube, while the second shows the different heights of the cube. Given that the system has three rows of cameras, in the simplest case, rows 1 and 4 of the table, we find three heights of the cube. An example of placing the cube at different heights can

Table 1: Mean positional (mm) and angular (degrees) errors for different heights and captures.

#Captures	#Heights	Errors (mm - degrees)					Total (mm - degrees)
		Row 1	Row 2	Row 3	Row 2-1	Row 3-2	
4	3	0.0014 - 0.63	0.0018 - 1.51	0.0013 - 3.11	0.0034 - 0.65	0.0035 - 4.04	0.0012 - 1.62
4	6	0.0015 - 0.45	0.0019 - 1.19	0.0012 - 1.27	0.0038 - 0.60	0.0030 - 3.10	0.0012 - 0.83
4	9	0.0015 - 0.44	0.0128 - 3.13	0.0147 - 3.73	0.0037 - 0.83	0.0553 - 12.64	0.0064 - 1.71
8	3	0.0013 - 0.44	0.0039 - 3.08	0.0041 - 3.39	0.0031 - 0.49	0.0138 - 12.28	0.0022 - 1.63
8	6	0.0015 - 0.42	0.0041 - 2.23	0.0042 - 2.37	0.0034 - 0.51	0.0142 - 8.31	0.0023 - 1.22
8	9	0.0015 - 0.37	0.0098 - 3.14	0.0112 - 3.69	0.0034 - 0.67	0.0409 - 12.80	0.0051 - 1.68

be seen in the figure 6. In the columns with the name Row 1,2, and 3, the results of the cube reconstruction for each row individually can be seen. The columns Row 2-1 and Row 3-2 refer to the results of reconstructing the cube using both rows at the same time

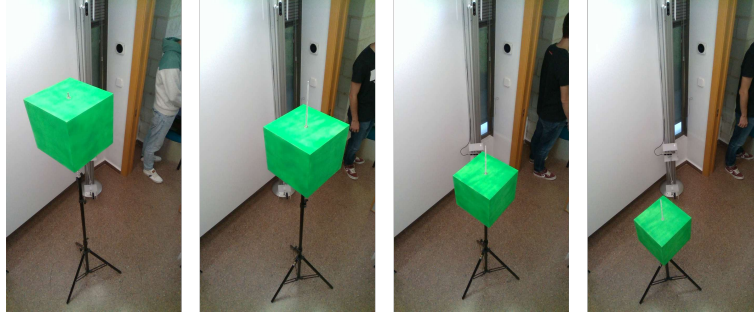


Fig. 6: View of the cube at different heights for a camera in the system during calibration.

In terms of the results, it can be observed that the configuration of 6 heights of the cube with 4 captures for each cube obtained the best results both in terms of reconstruction of the cube in millimeters and degrees obtained in the planes of the cube.

Using this configuration, we optimized the hyperparameters of our method by conducting a grid search involving 322 different configurations. The mean absolute error (MAE) was used as the loss metric to select the best hyperparameter configuration. In Table 2, the hyperparameters used can be observed. This optimization led to improved reconstruction accuracy. The error in millimeters has remained at the same value but the angular errors were reduced by 50%, from 0.83 to 0.41 degrees in the reconstruction of the cube using the transformation matrices obtained with calibration.

In Table 3, the results of calculating the Hausdorff and Wasserstein distances in 3D models of the human body reconstructed using different calibration settings can be observed. To perform these calculations, the optimized calibration that obtained the best results in the reconstruction of the cube was taken as ground truth. It can be seen how the other configurations, with the exception of the configuration of 3 heights

Table 2: Values of the hyperparameters used to obtain optimal calibration.

Parameter	Values
#Minimum captures for fitting the cube model	2, 3, 5
Maximum iterations	25, 50, 100
Distance threshold	0.003, 0.006, 0.01
Angular threshold	0.3, 0.6, 1
Distance threshold between rows	0.1, 0.001
Angular threshold between rows	5, 3, 2
Considering normals	True, False

and 4 captures, perform significantly worse. This is even more evident in the images shown in the qualitative results 7.

Table 3: Hausdorff and Wasserstein distances obtained from the optimized model and the non-optimized ones.

Configuration #Captures - #Heights	Distances	
	Hausdorff	Wasserstein
4 - 3	0.11	0.043
4 - 6	0.11	0.040
4 - 9	0.49	0.101
8 - 3	0.37	0.086
8 - 6	0.27	0.069
8 - 9	0.42	0.087

4.3 Qualitative Results

For the qualitative results in this work, we have reconstructed a human body using the calibration matrices obtained during quantitative experimentation. In Figure 7, the difference between the bodies reconstructed from the optimized and non-optimized calibration can be seen. In the central image, it is observed how from the optimized calibration the leg area aligns correctly with the rest of the body. While in the non-optimized one, a backward displacement of the lower limbs is observed due to calibration errors.

But having an optimal calibration not only improves point cloud registration but also enhances mesh generation and texture projection. As shown in Figure 8, the use of optimal calibration leads to cleaner texture projection with fewer visual artifacts.

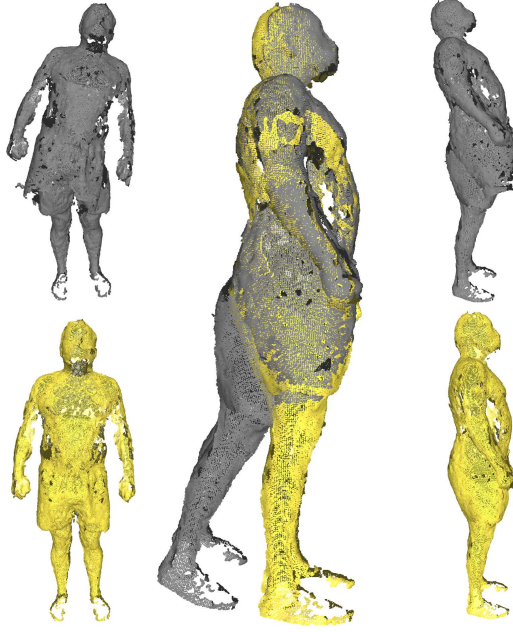


Fig. 7: Comparison of point clouds registered using the optimized calibration (yellow) and non-optimized (gray).

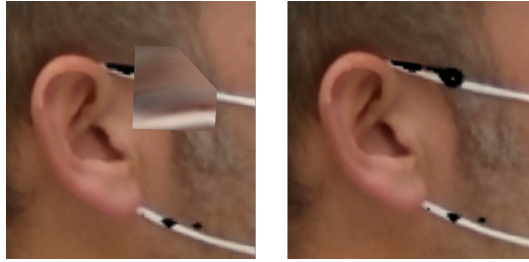


Fig. 8: Comparison of the texture generated from the model reconstructed with the non-optimized calibration (left) with the optimized one (right)

The proposed calibration method has been employed to reconstruct more than 400 human body models as part of the Tech4Diet project[18]. The dataset encompasses approximately 100 patients, each subjected to periodic clinical visits throughout their nutritional intervention, with an average of four visits per patient. During these visits, a full 3D body model was acquired following the methodology detailed in this work. Several examples of these reconstructions, achieved using the optimized calibration, are shown in Figure 9. Patients visualize their reconstructed body models via a Virtual

Reality application (Figure 10), which serves as an assistive tool to enhance adherence and engagement with their nutritional treatment.

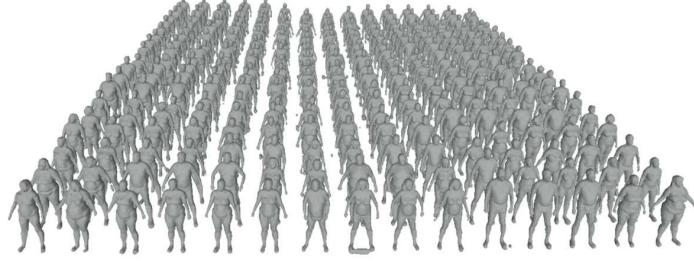


Fig. 9: Examples of 3D human body reconstructions obtained using the proposed calibration method within the Tech4Diet project.



Fig. 10: Visualization of a patient’s reconstructed body models through the Virtual Reality application, used as a supportive tool during nutritional interventions.

5 Conclusion

In this work, a marker-based multi-camera calibration method has been presented. The experiments performed demonstrate that the proposed method enhances the accuracy of 3D reconstruction by minimizing positional and angular errors in camera alignment. In particular, it has been observed that an optimal configuration of six marker heights with four captures per position produces the best results, reducing angular error by 50% after hyperparameter optimization.

A key innovation of the proposed approach is the use of a three-dimensional cube-shaped marker that ensures the visibility of multiple planes from a wide range of viewpoints, overcoming traditional limitations of planar or spherical markers. Unlike methods based on a single model fitting, such as those using spherical targets prone to ellipse fitting errors, our method fits three independent planes—one per visible

face—and applies strict orthogonality constraints between them. This multi-model fitting strategy significantly improves calibration robustness and enables highly accurate multi-camera alignment even under partial occlusions or suboptimal conditions.

Beyond this specific application, the developed methodology is extensible to other domains that require precise volumetric capture, such as body scanning in health and sports, avatar generation for virtual reality, film production, or ergonomic monitoring in industrial environments.

While the proposed approach has demonstrated promising results, several open challenges remain. Future work will focus on extending the method to other geometric configurations and validating its accuracy through comparisons between real-world anthropometric measurements and those obtained from the calibrated 3D reconstructions.

Declarations

- **Ethics Approval and Consent to Participate:** All participants provided informed consent to participate in the Tech4Diet project, agreeing explicitly to the use of their anonymized 3D reconstructions for research purposes.
- **Consent to Publish:** The participants provided informed consent to publish anonymized data and images resulting from the research. Explicit permission was obtained for publication purposes.
- **Data Availability Statement:** The datasets used during the current study, along with the implementation of our calibration method, are publicly available in the following GitHub repository: [Github](#).
- **Author Contribution:** Conceptualization and methodology, J.A.-L., A.F.-G.; software, J.A.-L., N.G.-D.; validation B.S.-S., A.M.-L.; supervision, H.M.-M.; writing—review, N.G.-D., J.A.-L. All authors have read and agreed to the published version of the manuscript.
- **Funding:** This work has been funded by the Spanish State Research Agency (AEI) through the grant PID2023-149562OB-I00 and PID2023-152804OB-I00, awarded by the MCIN/AEI/10.13039/501100011033, as well as by the consolidated group project CIAICO/2022/132 "AI4Health", financed by the Government of the Valencian Community.
- **Competing Interests:** The authors declare that they have no competing interests.

References

- [1] Park, B.-S., Kim, W., Kim, J.-K., Kim, D.-W., Seo, Y.-H.: Iterative extrinsic calibration using virtual viewpoint for 3d reconstruction. *Signal Processing* **197**, 108535 (2022) <https://doi.org/10.1016/j.sigpro.2022.108535>
- [2] Chaochuan, J., Ting, Y., Chuanjiang, W., Binghui, F., Fugui, H.: An extrinsic calibration method for multiple rgb-d cameras in a limited field of view. *Measurement Science and Technology* **31**(4), 045901 (2020) <https://doi.org/10.1088/1361-6501/ab48b3>

- [3] Tonchev, K., Manolova, A., Neshov, N., Poulkov, V.: Rgb-d sensors extrinsic calibration in controlled environment. In: 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), vol. 2, pp. 730–735 (2021). <https://doi.org/10.1109/IDAACS53288.2021.9660917>
- [4] Jing, Y., Yuan, C., Hong, X.: Online calibration between camera and lidar with spatial-temporal photometric consistency. *IEEE Robotics and Automation Letters* **9**(2), 1027–1034 (2024) <https://doi.org/10.1109/LRA.2023.3341768>
- [5] OSHIRO, A., AIHARA, K., JI, Y., TSUKIYAMA, K., AKAGI, T., BARILLARO, S.: One-shot calibration of rgb-d camera networks based on the novel design of cubic reference marker with unique shape features. *Journal of the Japan Society for Precision Engineering* **91**(1), 104–110 (2025) <https://doi.org/10.2493/jjspe.91.104>
- [6] Curto, E., Araújo, H.: 3d reconstruction of deformable objects from rgb-d cameras: An omnidirectional inward-facing multi-camera system. In: VISIGRAPP (2021). <https://api.semanticscholar.org/CorpusID:232094310>
- [7] Gao, J., Jiang, G., Gao, C.: A method for calibrating multi-camera systems based on sparse reconstruction of a 3d object. *Measurement* **240**, 115561 (2025) <https://doi.org/10.1016/j.measurement.2024.115561>
- [8] Dai, C., Han, T., Luo, Y., Wang, M., Cai, G., Su, J., Gong, Z., Liu, N.: Nmc3d: Non-overlapping multi-camera calibration based on sparse 3d map. *Sensors* **24**(16) (2024) <https://doi.org/10.3390/s24165228>
- [9] Shen, E., Hornsey, R.: Multi-camera network calibration with a non-planar target. *IEEE Sensors Journal* **11**(10), 2356–2364 (2011) <https://doi.org/10.1109/JSEN.2011.2123884> . Cited by: 40
- [10] Mehmandar, B., Talakoob, R., Poullis, C.: Neural Real-Time Recalibration for Infrared Multi-Camera Systems (2024). <https://arxiv.org/abs/2410.14505>
- [11] Dexheimer, E., Peluse, P., Chen, J., Pritts, J., Kaess, M.: Information-theoretic online multi-camera extrinsic calibration. *IEEE Robotics and Automation Letters* **7**(2), 4757–4764 (2022) <https://doi.org/10.1109/LRA.2022.3145061>
- [12] Dong, Y., Guo, W., Zha, F., Wang, P.: Extrinsic parameter calibration and dynamic objects coping strategy for multi-camera slam. In: 2022 3rd International Conference on Computer Science and Management Technology (ICCSMT), pp. 162–168 (2022). <https://doi.org/10.1109/ICCSMT58129.2022.00041>
- [13] Grammatikopoulos, L., Papanagnou, A., Venianakis, A., Kalisperakis, I., Sten-toumis, C.: An effective camera-to-lidar spatiotemporal calibration based on a simple calibration target. *Sensors* **22**(15) (2022) <https://doi.org/10.3390/>

- [14] Shin, J., Yun, S., Kim, A.: PeLiCal: Targetless Extrinsic Calibration via Penetrating Lines for RGB-D Cameras with Limited Co-visibility (2024). <https://arxiv.org/abs/2404.13949>
- [15] Yuesheng, H., Tao, W., Long, C., Hanyang, Z., Ming, Y.: An extrinsic calibration method for multiple infrastructure rgb-d camera networks with small fov. *IEEE Open Journal of Intelligent Transportation Systems* **5**, 617–628 (2024) <https://doi.org/10.1109/OJITS.2024.3361842>
- [16] Saval-Calvo, M., Azorin-Lopez, J., Fuster-Guillo, A., Garcia-Rodriguez, J.: Three-dimensional planar model estimation using multi-constraint knowledge based on k-means and ransac. *Applied Soft Computing* **34**, 572–586 (2015) <https://doi.org/10.1016/j.asoc.2015.05.007>
- [17] Azorin-Lopez, J., Sebban, M., Fuster-Guillo, A., Saval-Calvo, M., Habrard, A.: Iterative multilinear optimization for planar model fitting under geometric constraints. *PeerJ Computer Science* **7**, 691 (2021) <https://doi.org/10.7717/peerj-cs.691>
- [18] Tech4D: Tech4D: Artificial Intelligence for Temporal Modelling of Shapes, Signals and Data (2026). <https://tech4d.ua.es/>