Automatic silo axis detection from RGB-D sensor data for content monitoring

Oriol Vila [a,b,*], Imma Boada [a], Narcis Coll [a], Marta Fort [a], Esteve Farres [b]

[a] *Graphics and Imaging Laboratory, University of Girona, Girona, 17003, Catalonia, Spain*
[b] *Insylo Technologies S.L., Girona, 17003, Catalonia, Spain*

ARTICLE INFO

ABSTRACT

RGB-D sensors can be a low-cost solution for an accurate silo's content monitoring which is fundamental for its efficient management. Some reference information such as the position and orientation of the sensor with respect to the silo's geometry is fundamental for obtaining correct content measurements from acquired data. Since in real cases this information is not always known, a new method to obtain these measurements is proposed. This, taking as input sensor acquired data (represented as a point cloud), automatically computes the silo's axis to provide a new reference system from which the point cloud can be easily processed. The z-axis of this reference system coincides with the gravity axis and the xy-plane is parallel to the ground plane. It is obtained in a six-step process that exploits the silo geometry properties and an estimation of the shape tensors of the acquired points. The method has been implemented and tested on both synthetic and real silos, considering a complete silo's discharge process and different camera positions. Data acquired at each discharge has been transformed using the new reference system and compared with the silo's ground truth (manually obtained for the real silos). To evaluate the accuracy the input point cloud to adjusted point cloud average distance has been considered. In all the tests, the well-performance of the proposal has been demonstrated, achieving a maximum average distance error of less than 6 cm.

1. Introduction

In the last decades, technology has led to great changes in agriculture. The potential capability of the Internet of Things to connect devices and share information has transformed and complemented manual processes with fully or semi-automatic ones. These achieve more efficient and effective procedures with the purpose to align with current social and environmental needs. In this context, smart sensors are playing a fundamental role and different strategies have been proposed to control weather conditions, the progress of crop growth, the quality of the soil or the silos content, just to name a few (Tang et al., 2021; Tao et al., 2021; Idoje et al., 2021). Moreover, the low cost of sensors has opened the door to affordable solutions that are extensible to more scenarios and users. One of the key components of these solutions are machine vision systems.

Since the introduction of the Kinect sensor by Microsoft in 2010, RGB-D sensors have become an essential component of machine vision systems (Zhang, 2012) with many applications in agriculture (Fu et al., 2020). RGB-D cameras return images like an ordinary camera but in addition to color, each pixel value represents the distance to a point. To measure this depth value, different technologies can be used including structured light, time-of-flight, or active stereo vision (Giancola et al., 2018). RGB-D cameras are composed of: an infrared (IR) projector

that casts patterns or light pulses into the scene; an IR camera/s that captures the reflected IR light, and a color camera. Although the Kinect was originally conceived for gaming, its low cost, its reliability as well as its good documentation led the camera to be the basis of many innovative applications in the robotic and computer vision community (Han et al., 2013). Moreover, the sensor has been the basis of other RGB-D cameras such as Asus Xtion (Anon, 2022c), Orbbec Astra 3D (Anon, 2022b), Occipital Structure Sensor (Anon, 2022d) or Intel RealSense D415 (Anon, 2022a).

In this paper, we will focus on the use of RGB-D cameras for the control of silos content which is fundamental for its proper management (Raba et al., 2020; Camps et al., 2019). Inaccurate measurements can hinder the management and optimization of supply chain logistics, leading to inefficiencies and increased costs as described in Raba Sánchez (2021).

1.1. Related work

Low-cost 3D sensors have become an emerging focus of research and development and a promising and cost-effective solution for many high-accuracy applications. In the context of electronics industry, Dimitriou et al. (2019) proposed a fault diagnosis system to inspect glue

---

* Corresponding author at: Graphics and Imaging Laboratory, University of Girona, Girona, 17003, Catalonia, Spain.
E-mail addresses: oriol.vila@insylo.com (O. Vila), imma.boada@udg.edu (I. Boada).

dispensation and die attachment on printed circuit boards. Their system automates fault diagnosis by accurately estimating the volume of glue deposits before and even after die attachment using a modular scanning system that produces high-resolution point clouds whereas the actual estimation of glue volume is performed by Regression-Net, a three-dimensional convolutional neural network. Later, the same authors in Dimitriou et al. (2020) proposed an architecture based on this network to model the geometric variations in manufacturing parameters and predict upcoming events related to sub-optimal performance. The method was also validated on a microelectronics use-case. In the context of cultural heritage preservation, Rousopoulou et al. (2019) proposed an integrated, portable solution based on a modular architecture, for accurate multi-sensorial 3D scanning via a dedicated motorized mechanical arm and efficient automatic 3D reconstruction of a big variety of cultural heritage assets even in situ. The key strength of their system is that it is a cost-effective and time-saving solution. Focused on automated 3D modeling of indoor spaces, Ingman et al. (2020) compared the performance of three low-cost sensor systems; one RGB-D camera, one low-end terrestrial laser scanner (TLS), and one panoramic camera, using a cloud-based processing service to automatically create mesh models and point clouds, evaluating the accuracy of the results against a reference point cloud from a higher-end TLS. They concluded the TLS performed the best both in terms of reconstructing the overall room geometry and smaller details, with the panoramic camera clearly trailing the other systems and the RGB-D offering a middle ground in terms of both cost and quality. Centering on simultaneous localization and mapping (SLAM), Liu et al. (2019) proposed an effective solution for RGB-D based SLAM by integrating an Inertial Measurement Unit into a recurrent and convolutional neural network that leads to enhanced pose estimation and point cloud registration. Kolhatkar and Wagle (2020) reviewed the different techniques used in mapping and localization of mobile robots and designing low-cost mobile platforms with sensors like RPLIDAR and Microsoft Kinect. Sun et al. (2020) proposed an innovative background modeling method by using both the color and depth information from an RGB-D camera. Their method, evaluated using a public RGB-D data set, was able to achieve superior performance compared with existing well-known methods. Park et al. (2020) proposed a deep learning-based mobile augmented reality (AR) for intelligent task assistance by conducting 3D spatial mapping without pre-registration using AR markers, which can match virtual AR objects to their corresponding physical objects automatically and accurately using single snapshot-based RGB-D data. In all these applications a rigorous calibration and error modeling of RGB-D camera data to produce high-quality information is required (Darwish et al., 2017; Liu et al., 2020). Regarding calibration techniques, Darwish et al. (2017) presented a new model for calibrating the depth measurements of RGB-D sensors based on the structured light concept. They also proposed an automatic method for the calibration of all RGB-D parameters, including internal calibration parameters for all cameras, the baseline between the infrared and RGB cameras, and the depth error model. When compared with traditional calibration methods, their approach showed a significant improvement in depth precision for both near and far ranges. More recently, Liu et al. (2020) proposed a novel and accurate sphere-based calibration framework for estimating the intrinsic and extrinsic parameters of color-depth sensor pair. They also proposed and analyzed a method of depth error correction.

1.2. RGB-D cameras for the control of silos

There are several technologies that can be used to determine the material level or amount of bulk solids within a storage silo. Some of them were reviewed in Lewis (2004), where is concluded that the most precise solutions are also the most expensive. However, the advent of low-cost 3D sensors provides a promising and cost-effective solution for this task. In the context of RGB-D cameras for silo's content measurements, few proposals have been advanced to date. Raba et al.
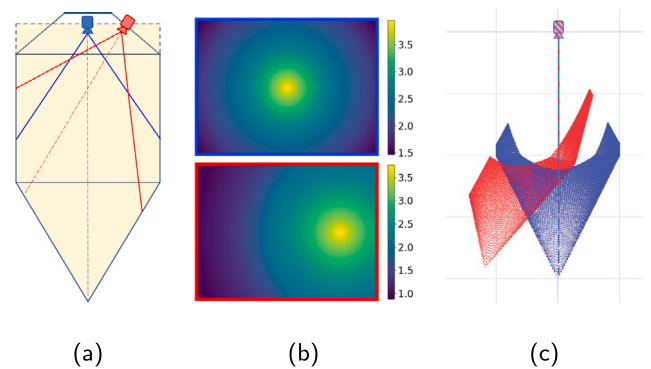

Fig. 1. (a) A silo with two different cameras, the blue one in the ideal position (in a coordinate system with *z*-axis coincident with gravity axis and XY-plane parallel to the ground) and the red one in the common position; (b) Depth maps obtained from the cameras with a blue/yellow color scale and expressed in meters, and (c) the obtained point clouds. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(2020) considered the use of RGB-D cameras for estimating silo volumetric measurements which provided unmatched accuracy compared to single-point level sensors.

Note that in this use case, the camera placement is decisive for the content measurement. Particularly, if the camera is placed at the center of the silo and completely perpendicular to the ground, the acquired data represent the distance between the silo content and the image plane of the camera, and hence the silo content can be easily computed. Unfortunately, in the majority of cases, this ideal placing is not accessible, due to silo features or installation restrictions, and specific processing of acquired data is required to measure the silo's content (see Fig. 1). To solve similar problems, point cloud registration, gradient-based optimization or even deep learning pose estimation techniques have been proposed. For instance, Holz et al. (2015) proposed a point cloud registration framework based on the Iterative Closest Point (ICP) algorithm widely used in the alignment of 3D structures. The main limitation of this approach is that it is necessary to know the silo's geometry to obtain the target point cloud to align with. In Fu and Hu (2012) the theoretical aspect of perturbation analysis via conditioning with a broad range of applications of perturbation analysis are presented. To be applied to our case, both the geometrical parametrization of the silo as well as the camera position serve as input variables which leads to high dimensionality with a large solution space. The high computational costs required to avoid local minima and reduce the dispersion in the solutions is the main limiting factor. Regarding deep learning solutions, Kehl et al. (2016) proposed a method for 6D pose estimation in 3D objects using locally sampled RGB-D patches and a convolutional auto-encoder. In this approach, the required training process that adds complexity to the method and the requirement of a custom dataset are the main limiting factors.

To overcome all these limitations, in this paper, a method to automatically obtain the new reference system and the adjusted point cloud is proposed. This approach is tailored to the measuring environment by exploiting the fundamental geometrical properties of the grain silos to ensure a reliable and simple approach that does not require prior information on the geometry of the silo. Furthermore, the method does not require installation parameters to be known nor specific calibration techniques. It takes as input the acquired RGB-D data and returns the coordinate system transformation parameters to represent acquired data in this new reference system, with the *z*-axis coincident with the gravity axis and the xy-plane parallel to the ground plane. To test and show the well-performance of the proposal, synthetic and real silos with different geometries, content states, and camera positions have been considered.
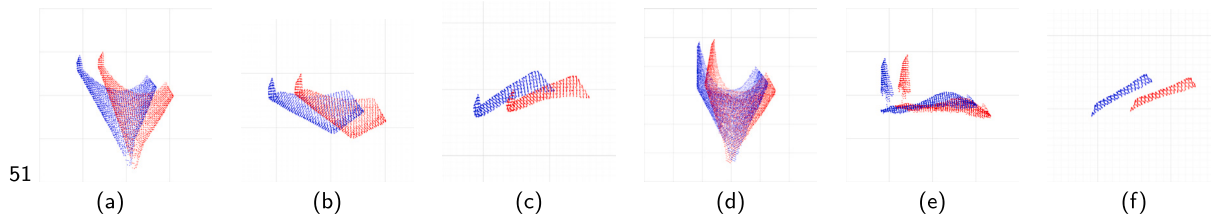
Fig. 2. Real point clouds acquired with a 3D sensor (in red) and the adjusted ones (in blue), (a)–(c) $Silo_{R1}$, (d)–(f) $Silo_{R2}$ of Table 2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
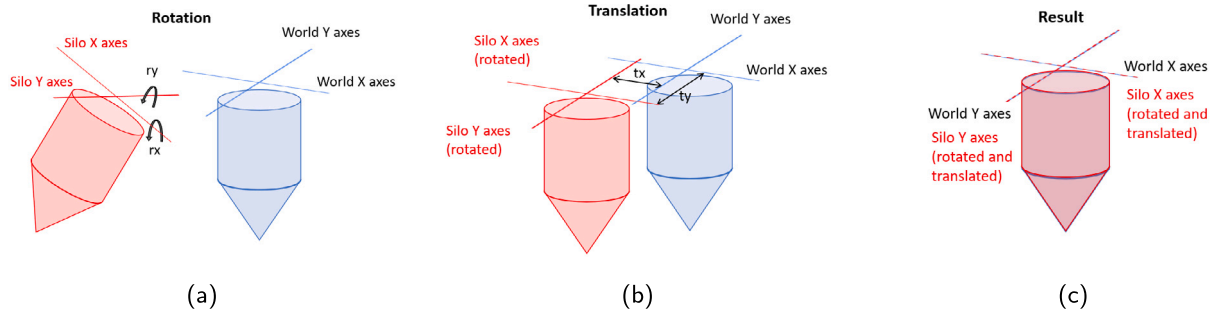

Fig. 3. The transformation procedure to position the input point cloud captured by the camera (represented in red) into the desired position (represented in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Besides this introduction, the paper has been structured as follows. In Section 2, the proposed adjustment approach is presented with a detailed description of the theoretical framework behind the proposal. The real and synthetic datasets used for the testing process are also described as well as the different experiments that have been carried out. Obtained results are presented and discussed in Section 3. Finally, conclusions and future work are given in Section 4.

2. Materials and methods

2.1. Problem description

To describe the problem a RGB-D camera placed at the top of the silo acquiring depth maps from its interior has to be considered. For each acquisition, a depth map, **D**, is obtained and a point cloud **P** is generated by directly assigning the depth value of each pixel to the z−coordinate and calculating the x− and y−coordinates using Eqs. (1) and (2) (see Fig. 2). In these, f_x and f_y are the intrinsic camera parameters corresponding to the horizontal and vertical focal lengths and c_x, c_y are the principal point of the camera, both expressed in pixels.

$$x = \frac{z}{f_x} \cdot (u - c_x) \qquad (1)$$

$$y = \frac{z}{f_y} \cdot (v - c_y) \qquad (2)$$

As shown in Fig. 2 the 3D reconstructions obtained from the depth maps using the camera coordinate system (represented in red), do not match the silo boundaries defined in the reference system (represented in blue). To transform acquired points to the silo reference coordinates system, it is necessary a rotation plus translation adjustment (see Fig. 3). The challenge is how to automatically compute this transformation taking into account the variability of situations that can appear due to both silos and content shapes.

2.2. Theoretical framework

To tackle the problem, we will focus on the mathematical properties of the points that are part of the silos walls. To extract these properties the shape tensor needs to be introduced.

2.2.1. The shape tensor

The shape tensor (or the second fundamental form) is a mathematical tool used to measure the intrinsic curvature of a surface, i.e., a way of measuring in a neighborhood of a point on a surface how fast the surface moves away from the tangent plane to the surface at the point. Its mathematical formulation is based on the concepts of normal section and normal curvature.

Being p a regular point on a surface S (all tangents to S through p lie on a plane), let N and T be the unitary normal vector and a unitary tangent vector of S at p, and $\pi(p, T, N)$ the plane defined by p, T and N. The curve resulting from the intersection of S with $\pi(p, T, N)$ is the *normal section* of S at p along T, and its curvature is the *normal curvature* of S at p along the direction of T (see Fig. 4).

The *shape tensor* of S at p is the function $K_p(T)$ that measures the normal curvature of S at p along the direction of T. If $\{T_1, T_2\}$ is an orthonormal basis of the tangent plane to S at p, the tensor of curvature can be expressed by the quadratic form

$$K_p(T) = \begin{pmatrix} t_1 & t_2 \end{pmatrix} \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \end{pmatrix},$$

where $T = t_1 T_1 + t_2 T_2$.

Such a function $K_p(T)$ has two extreme values, $k_1 \geq k_2$, called the *principal curvatures* of S at p. Their associated tangent vectors, $\{E_1, E_2\}$, are called the *principal directions* of S at p. It can be proved that k_1, k_2 are the eigenvalues of the $K_p(T)$ matrix. Besides, $\{E_1, E_2\}$ is an orthonormal basis that can be computed from the $K_p(T)$ matrix eigenvectors (e_{11}, e_{21}) and (e_{12}, e_{22}) by $E_1 = e_{11} T_1 + e_{21} T_2$ and $E_2 = e_{12} T_1 + e_{22} T_2$. See do Carmo (1976) for full details and to find out how the shape tensor can be computed on parametric surfaces.

2.2.2. The shape tensor on a cylinder or a cone

Once the shape tensor has been introduced let us analyze it on the geometric structures defining our silos. In the following, S, will represent a silo's surface, i.e., the union of a cylinder and a cone, $\mathcal{A}$ its axis, X the direction vector of $\mathcal{A}$, p a point on S (different from the cone's vertex), and R the cylinder's radius.

Note that the cylinders and cones defining silos have the (revolution) axis perpendicular to the basis, and the cylinder generatrix parallel to its axis. For the sake of simplicity, we talk about cylinders
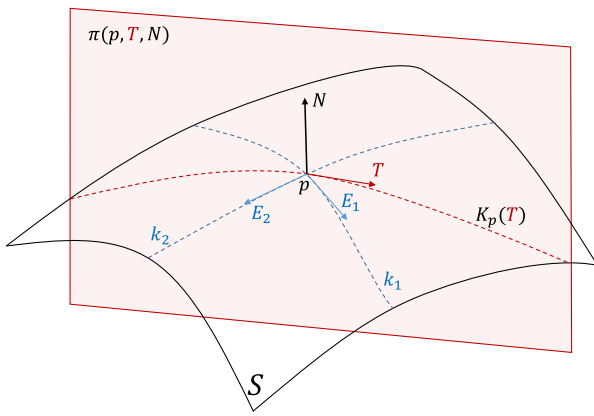
Fig. 4. Normal sections and principal curvatures.

and cones, but they are right cylinders and right cones. Due to their rightness, the following well-known properties hold.

Property 1. *For any p on S, $k_2 = 0$ and E_2 is the direction vector of the generatrix of S.*

Since a straight-line normal section always exists, its normal curvature is 0 and hence $k_2 = 0$ and E_2 is its director vector.

Property 2. *Given a point p on S the plane $\pi(p, E_2, N)$ contains $\mathcal{A}$, the surface revolution axis.*

This plane contains the generatrix and the normal vector, therefore it also contains the axis.

Property 3. *Given two points p and p' on S, the axis $\mathcal{A}$ is parallel to $E_1 \times E_1'$, i.e., $X = E_1 \times E_1'$.*

The principal direction E_1 is orthogonal to E_2 and N, and according to Property 2 it is also orthogonal to X. Since E_1 and E_1' are both orthogonal to X, X is their cross-product.

Property 4. *Being p a point on a cylinder, its principal direction E_2 is the direction X of the cylinder's axis.*

The cylinder's generatrix and the axis are parallel, hence $X = E_2$.

Property 5. *Given a point p on the cylinder, $k_1 = 1/R$.*

The normal sections of a cylinder at p change from a circle perpendicular to the axis to the generatrix, passing through a set of ellipses. Hence, its maximal curvature k_1 is the curvature of the circular section which is $1/R$.

Note that Properties 4 and 5 only hold on cylinders, they are not true on cones.

2.2.3. Silo's characterization

The silo surface can be partitioned into three regions that can be fully characterized as follows:

- $\mathcal{R}_0$ composed of non-regular points, i.e., the con vertex, the cylinder–cone connecting zone and the cylinder end. For these points the tangent plane does not exist and the shape tensor is not defined.
- $\mathcal{R}_1$ are points of the cylinder interior for which $k_2 = 0$, $E_2 = X$ and k_1 is constant in a neighborhood and equal to $1/R$. Due to Properties 1, 4 and 5.
- $\mathcal{R}_2$ are points of the cone interior for which $k_2 = 0$ and k_1 is not constant in a neighborhood. Due to Properties 1 and Properties 5.

If the silo is empty, $\mathbf{P} = \mathcal{R}_0 \cup \mathcal{R}_1 \cup \mathcal{R}_2$, otherwise a *feed region*, $\mathcal{R}_3$, corresponding to the feed surface has to be added.

Moreover, two points p and p' on $\mathcal{R}_1 \cup \mathcal{R}_2$ fully characterize the silo's axis. Due to Properties 2 and 3, Eqs. (3) and (4) hold:

$$X = E_1 \times E_1' \tag{3}$$

$$\mathcal{A} = \pi(p, X, N) \cap \pi(p', X, N') \tag{4}$$

2.3. The adjustment algorithm

The proposed algorithm applies the presented theoretical framework to compute the silo's axis taking as input a point cloud $\mathbf{P}$ obtained from a RGB-D camera. Different to the ideal case previously described, our input data will be a discretization of the ideal case. Moreover, in case of a real sensor, such data will have noise due to temperature, humidity, and other phenomena. Therefore, to obtain robust results it is necessary to consider several points and average measures in order to determine the silo's axis and its direction.

The proposed algorithm is composed of the following steps:

1. Pre-processing of input data. Depth map points are transformed to real world coordinates and downsampled using a $0.05\ \mathrm{m} \times 0.05\ \mathrm{m} \times 0.05\ \mathrm{m}$ grid. This will led to a spatial uniform distribution and a reduction on the total number of points decreasing computational costs.

2. Normal vectors and shape tensors computation. The normal vector of each input point p is computed by the method presented by Hoppe et al. (1992). That is, given k points p_i in a local neighborhood of p, the normal of p is computed as the smallest eigenvector of the matrix

$$M = \frac{1}{k} \sum_{i=1}^{k} (p_i - p) \cdot (p_i - p)^T \ .$$

Then, the discrete shape tensor is calculated using the normal components of the point and its local neighborhood according to the adjacent-normal cubic approximation method described in Goldfeather and Interrante (2004).

3. Selection of candidate points. To reduce the number of points required for the computations while increasing the ratio of points that belong to the silo structure a subset of input point cloud is selected. Particularly, we select a subset of points with the minimum k_2 component, i.e., points that potentially belong to $\mathcal{R}_1 \cup \mathcal{R}_2$ and not to $\mathcal{R}_3$.

4. Determination of cylinder and cone axis direction X. By applying Eq. (3) to each pair of candidate points a collection of estimated vectors $\widetilde{X}$ is obtained. From this, the direction of maximum variability is computed using principal component analysis (PCA) decomposition. Such a direction corresponds to the cylinder and cone axis direction and defines the rotation applied to the input points at Step 6 to get the position represented in Fig. 3(a).

5. Determination of a point a on the cylinder and cone axis $\mathcal{A}$ at the camera world-height. We compute a collection of estimated points on the axis by using Eq. (4) over a set of pairs of candidate points. As the set of all possible pairs between candidate points is very large, for each candidate point p we select s candidate points p' satisfying that the dot product between $N \times X$ and N' is in the 1st quartile. For each pair (p, p'), we intersect the plane $\pi(p, X, N)$ with the normal line $p' + \lambda N'$ obtaining an estimated point on the axis. Then, we project all the estimated points into the plane perpendicular to X containing the origin and obtain a using the median operator. Since the origin is the camera and the plane is perpendicular to X, the plane is parallel to the floor and the projected points and a are at the camera world-height. Point a leads to the translation to apply at step 6 to get the matching of Fig. 3(b).
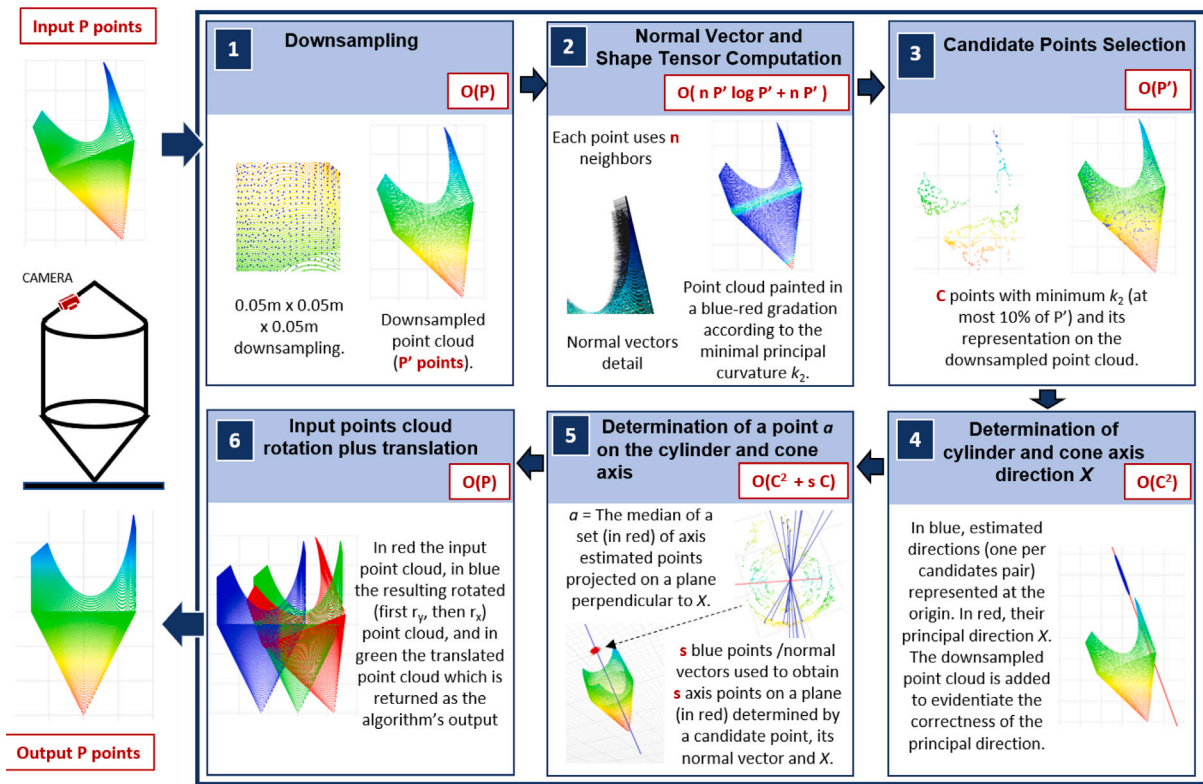
Fig. 5. A schema of the algorithm steps with summaries, images, and the computational costs.

6. Adjustment of input points via rotation plus translation. The rotation and translation that brings $\mathcal{A}$ to the z-axis is applied to input points obtaining the adjusted data for proper computations. In order to perform this transformation, we can perform a change of basis or apply a sequence of rotation and translation operations that can be defined in a single transformation matrix (see Eq. (5)). Because the structure of a silo (wall of a silo) presents revolution symmetry, it is possible to define the rotation with just two rotation components: rotation on the x-axis (r_x) and rotation on the y-axis (r_y). These rotation components can be obtained from $X = (x, y, z)$ by using the formulas of Eq. (6). The final translation can be directly obtained from point $a' = (x_{a'}, y_{a'}, z_{a'})$ being a' the rotations applied to the point a. Because we decided to respect the height of the camera placement, the translation is described by the component in the x-axis (t_x) and the component in the y-axis (t_y) according to Eq. (7).

$$\begin{bmatrix} \cos(r_y) & \sin(r_x)\sin(r_y) & \cos(r_x)\sin(r_y) & t_x \\ 0 & \cos(r_x) & -\sin(r_x) & t_y \\ -\sin(r_y) & \sin(r_x)\cos(r_y) & \cos(r_x)\cos(r_y) & 0 \end{bmatrix} \quad (5)$$

$$r_x = \arcsin\left(y/\sqrt{1-x^2}\right) \quad r_y = \arcsin(-x) \quad (6)$$

$$t_x = -x_{a'} \quad t_y = -y_{a'} \quad (7)$$

In Fig. 5 the information obtained at each step of the algorithm is illustrated.

2.4. Testing scenarios

In this section, the synthetic and real testing scenarios, procedures, and sensors that have been used to evaluate the performance of the algorithm are presented.

2.4.1. Perfect synthetic scenario

The perfect synthetic scenario simulates a silo with a perfect geometry in a controlled discharge sequence with a known feed distribution and shape. The feed has been modeled as a Gaussian-shaped surface that, in the orthonormal reference system centered at the vertex of the silo and third vector parallel to the axis of the silo, is given by

$$z = H - 0.75R\left(1 - e^{-\frac{x^2+y^2}{R^2}}\right),$$

where H is the maximum height of the feed with respect to the vertex of the silo and R is the silo's cylinder radius. The camera position and orientation are also known and hence the exact values of rotation and translation that have to be applied to obtain the point cloud in the reference coordinate system can be computed. The three configurations collected in Table 1 have been considered: $Silo_1$, is small silo, with a proportionally short cone and a camera placed very close to the center with a slight rotation on the camera so that the camera points towards the con vertex; $Silo_2$, is a bigger silo with a camera placed near the cylinder wall and with a more severe rotation such that the camera points towards the vertex's cone; and $Silo_3$, is a silo with a short cylinder but a large cone and with the camera rotated and translated in both axes such that the field of view only captures a small portion of the cylinder walls. Note that, $Silo_1$ and $Silo_2$ aim to reproduce the common placement of real cases where the maximum field of view of the inner silo is desired while $Silo_3$ recreates an undesired situation.

For each setup configuration, we have evaluated a full discharge sequence considering the feed level ranging from the maximum height to the minimum in 100 steps. For each step, we have generated a synthetic depth map using the camera intrinsic parameters, the position according to the translation and rotation values, and the geometrical parametrization of the silo structure (see Fig. 6). The 100 depth maps serve as input of the algorithm, while the reference parameters serve as the ground truth for the performance evaluation.

Table 1
Geometrical specifications and reference adjustment parameters for each synthetic silo.

	$Silo_1$	$Silo_2$	$Silo_3$
Cylinder Height (m)	4	4	1
Cone Height (m)	1	3	3
Radius (m)	1	1.5	1.25
Translation X (t_x) (m)	0.25	1.25	0.75
Translation Y (t_y) (m)	0	0	−0.75
Rotation X (r_x) (rad)	0	0	0.15
Rotation Y (r_y) (rad)	0.15	0.35	0.25

(a)   (b)   (c)

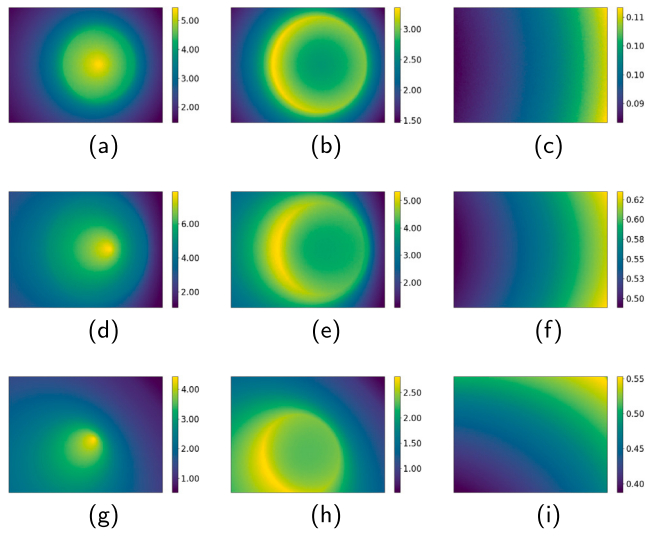(d)   (e)   (f)

(g)   (h)   (i)

Fig. 6. Generated synthetic depth maps for different feed levels corresponding to $Silo_1$ (a–c), $Silo_2$ (d–f), and $Silo_3$ (g–i). To enhance the visualization of the shape in every depth map, we have applied a color map that spans from the minimum (blue) to the maximum (yellow) depth value in meters. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2.4.2. Perturbed synthetic scenario

Despite being widely used in robotics and computer vision, a main limitation of low-cost cameras is the acquisition error caused by their physical characteristics and algorithms used for their measurements (Cabrera et al., 2018). An error that, in case of large range acquisition, can still be more relevant. The perturbed synthetic scenario simulates this error by considering a new set of synthetic perturbed depth maps with the same geometrical specifications of Table 1. To create the maps, the error of the Orbect Astra S (similar to the Orbbec Astra used in our real scenario) has been used. Following Giancola et al. (2018) the error has been modeled by a normal distribution with mean $s(d)$ (see Eq. (8)) and standard deviation $r(d)$ (see Eq. (9)). Note that this error measurement is quadratic with respect to the measured distance. Synthetic perturbed depth measurements are obtained from Eq. (10) where d' is the real measured distance and $e(d')$ is the error on the given distance. Fig. 7 shows a perturbed depth map for $Silo_2$ of Table 1.

$$s(d) = (2.2093E - 5)d^2 - 0.017878d + 3.4397 \text{ [mm]} \qquad (8)$$

$$r(d) = \frac{(3.E - 6)d^2 - (5.E - 4)d + 0.5}{2\sqrt{3}} \text{ [mm]} \qquad (9)$$

$$d = d' + e(d') \qquad (10)$$

The impact of this error is analyzed with the same silos and discharge sequences of Section 2.4.1.

2.4.3. Real scenario

Two real silos with different geometries and camera placements have been considered. For each silo, we have captured a complete discharge with one camera acquisition per hour obtaining 296 depth
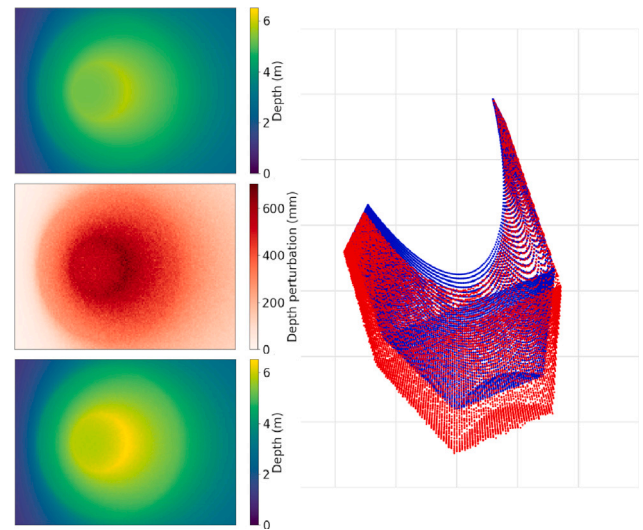

Fig. 7. The left column shows the exact depth map, the error corresponding to each point, and the perturbed depth map for $Silo_2$. The right column shows the exact synthetic silo (in blue) and the perturbed one (in red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Geometrical specifications and the reference adjustment parameters (manually obtained) for each real silo.

	$Silo_{R1}$	$Silo_{R2}$
Cylinder Height (m)	2.75	1.4
Cone Height (m)	2	1.85
Radius (m)	1.18	1.07
Translation X (t_x) (m)	0.7	0.54
Translation Y (t_y) (m)	−0.04	0.01
Rotation X (r_x) (rad)	0.013	0.1
Rotation Y (r_y) (rad)	0.16	0.1

maps for $Silo_{R1}$ and 418 for $Silo_{R2}$ (see Fig. 8). Note that, different to synthetic case, external conditions such as temperature, camera errors, or calibration, can introduce noise and lead to errors on the 3D reconstructions.

For real silos, when low-cost cameras are used, and contrarily to the synthetic scenarios, the exact camera position and orientation are not known and a trial-and-error manual process has been applied to obtain them. This process applies rotations and translations to several point clouds and inspects the results until a visually satisfactory approximation is found. The parameters obtained by this process are presented in Table 2 and considered the ground truth.

2.4.4. Testing procedure

For the tests, a complete discharge process has been considered and, for most of the cases, no silo's wall was visible in the initial images. Since the accuracy of the results depends on the silo's wall visibility, the algorithm has been applied to every step of the discharge captured by the camera, providing a set of estimated parameters. At each new step, more accurate parameters have to be expected since more silo's wall has been captured. Note that, there can be some steps of the discharge where there is no output parameters, because the wall visibility is too small and the algorithm did not have enough candidate points to provide a result.

In the perfect synthetic scenario, the last estimated set of parameters provided by the algorithm has been used as the adjustment result. Whereas, in the perturbed synthetic and real scenarios, in order to obtain the adjustment results for each silo, we have averaged the last 10 sets of estimated parameters to reduce the noise fluctuations. To evaluate the relationship between silo's wall visibility and algorithm
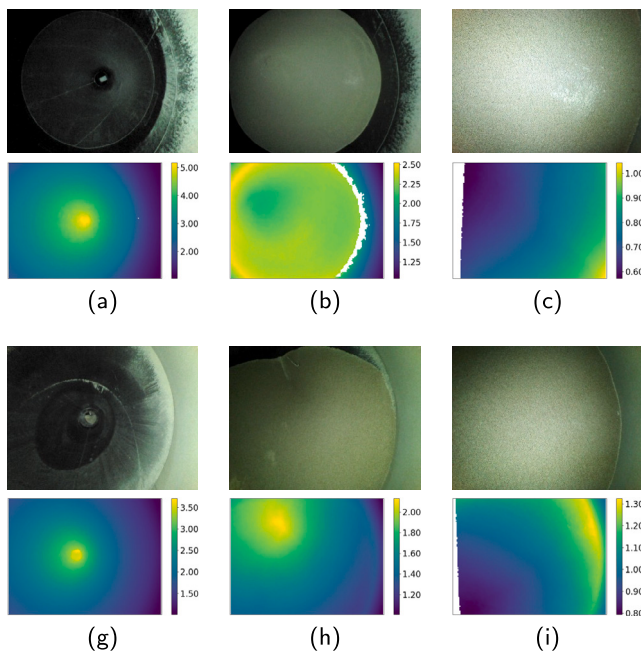
(a) (b) (c)

(g) (h) (i)

Fig. 8. The RGB and depth map captures for different feed levels corresponding to the $Silo_{R1}$ (a–c) and $Silo_{R2}$ (g–i). Depth value represented in meters using a color map. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

accuracy, the *cylinder, cone* and *feed ratios* have been computed. These are defined as the relationship between the number of depth map points belonging to the cylinder, the cone and feed compared to the total number of points with depth information, respectively. The error of the adjustments has been computed considering the difference between the estimated transform (translation plus rotation) and the ground truth transform obtained from the camera position. The last is known for the synthetic case, while for the real one has been manually obtained. Finally, to measure the goodness of the adjustment, the PC average distance, denoted by PC average distance along the paper, has been used. This indicates how well is an adjustment by measuring the average distance of the points between the ground truth adjustment and the estimated one. It is computed from Eq. (11) where $P_a(n)$ is a point of the adjusted point cloud and $P_r(n)$ is the closest point of the reference point cloud to $P_a(n)$, i.e. at lowest distance.

$$Dist = \frac{1}{N} \cdot \sum_{n=1}^{N} P_a(n) - P_r(n) \qquad (11)$$

2.4.5. Sensor comparison

Measurements can differ between sensor technologies due to their accuracy and camera calibration features. To evaluate this effect on the algorithm performance a new empty real silo and four different sensors (the structured light Orbbec Astra, the active stereo Intel RealSense D415 and Intel RealSense D455, and the LIDAR sensor Intel RealSense L515) have been considered. The silo's geometry (manually obtained) is 1.65 m cylinder height, 1.7 m cone height, and 0.78 m radius. Each sensor has been evaluated independently. It has been placed on a reference zone at the top side of the silo in such a way that part of the cone wall is visible. Due to sensor features, the rotation will be different for each camera. Once placed, each sensor captured a depth map that will be processed using the proposed algorithm to obtain the silo axes and the transformations to be applied.

A synthetic point cloud generated from the silo geometrical measurements has been used to measure the point cloud-to-point cloud distance. In addition, for the Intel Real Sense D455 and L515 use cases, the accelerometer of these devices has been used as the camera orientation ground truth.

Table 3

For each perfect synthetic silo, the error on t_x and t_y and the point cloud average distance (PC average distance) in mm, and r_x and r_y in *mrad* on the stabilized zone.

Measured error	$Silo_1$	$Silo_2$	$Silo_3$
Translation X (t_x) (mm)	0.61	−0.01	−0.40
Translation Y (t_y) (mm)	−0.02	0.57	0.67
Rotation X (r_x) (mrad)	−0.01	0.13	0.33
Rotation Y (r_y) (mrad)	−0.39	0.06	−0.25
PC average distance (mm)	0.66	0.32	0.96

3. Results and discussion

The proposed approach has been implemented in Python 3.7.4 and tested on a laptop with an Intel Core i7-7700HQ as a CPU and memory of 16 GB DDR4 at 2400 MHz. To present the results of our tests, first, we are going to evaluate the synthetic cases and then, the real one. Afterwards, some general considerations will be presented taking into account both situations.

3.1. Perfect synthetic scenario evaluation

To evaluate the algorithm convergence, in Fig. 9(a–c), for each perfect synthetic silo, the algorithm estimated parameters errors (i.e. the error in r_x, r_y, t_x, and t_y) at different discharge steps are plotted. In addition, to assess the relationship between the algorithm convergence and the distribution of points in the depth map, in Fig. 9(d–f), the cylinder, the cone, and the feed ratios with the PC average distance are plotted. For all the cases, the PC average distance improves as the cone plus cylinder ratio increases up to the point of convergence. Note that, to achieve the maximum accuracy the visibility of an empty silo is not required. For all tested silos, the algorithm results became stable after the feed level reached 35% of the total height of the silo.

For each synthetic silo, Table 3 shows the adjustment values when the algorithm results stabilized with an average PC average distance of 0.65 mm (see PC average distance in the table). Hence, the average adjustment inaccuracy of the algorithm achieved less than 1 mm of error. Indeed, all silo adjustment results converged at the same pace and achieved sub-centimeter accuracy once the feed ratio was below 50%. Therefore, the different silo geometries and the different camera positions did not have a great impact on the convergence speed nor on the accuracy of the algorithm when processing exact data. To visually illustrate these results, in Fig. 10, for each one of the synthetic silos, the correct adjustment (in blue) and the estimated with the proposed adjustment algorithm after convergence (in red) are presented.

3.2. Perturbed synthetic scenario evaluation

To evaluate the algorithm convergence, in Fig. 11(a–c), for each perturbed synthetic silo, the algorithm estimated parameters error (i.e., the error in r_x, r_y, t_x, and t_y values) at different discharge steps are plotted. In addition, to assess the relationship between the algorithm convergence and the distribution of points in the depth map, in Fig. 11(d–f), the cylinder, the cone, and the feed ratios with the PC average distance are plotted. It can be observed that the differences in the rotation and translation values, as well as the PC average distances, are no longer converging to zero. They exhibit more significant fluctuations due to the added random error. Additionally, these values increase with the feed discharge, as the camera progressively captures further distances that are subject to more measurement errors, thereby distorting the geometry. Note that the algorithm provides a silo axis different than the one used to generate the data. Moreover, the axis error is proportional to the feed amount and the camera rotation.

Table 4, for each perturbed synthetic silo, shows the adjustment values when the algorithm results stabilized with a PC average distance of 14.06 mm, 62.55 mm, and 27.67 mm, respectively. As expected,
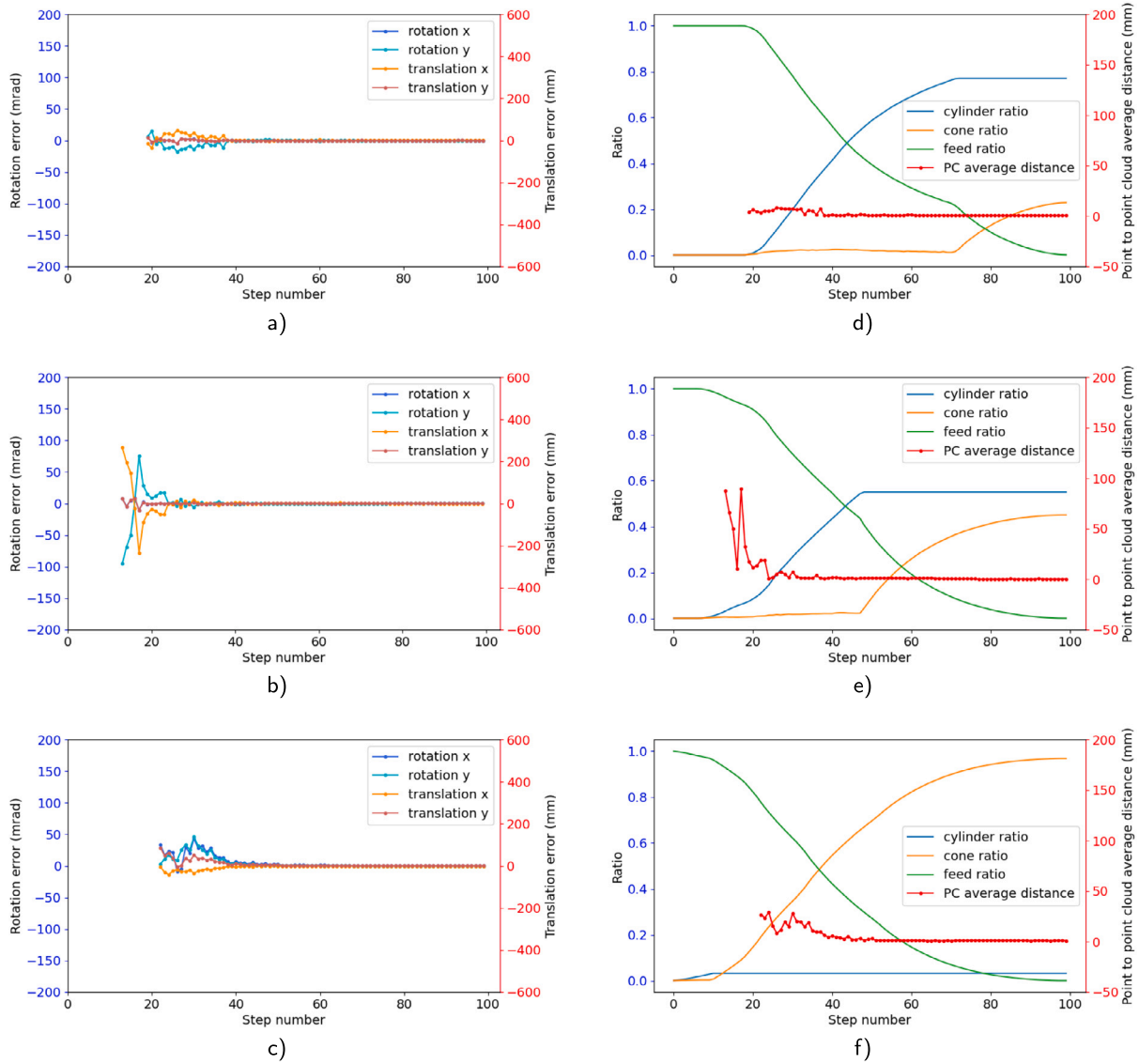
Fig. 9. From (a) to (c), the algorithm output error, i.e., t_x, t_y, r_x, and r_y, at different steps of the algorithm for the perfect $Silo_1$, $Silo_2$ and $Silo_3$ respectively. From (d) to (e) the cylinder, the cone, and the feed ratios with the PC average distance.
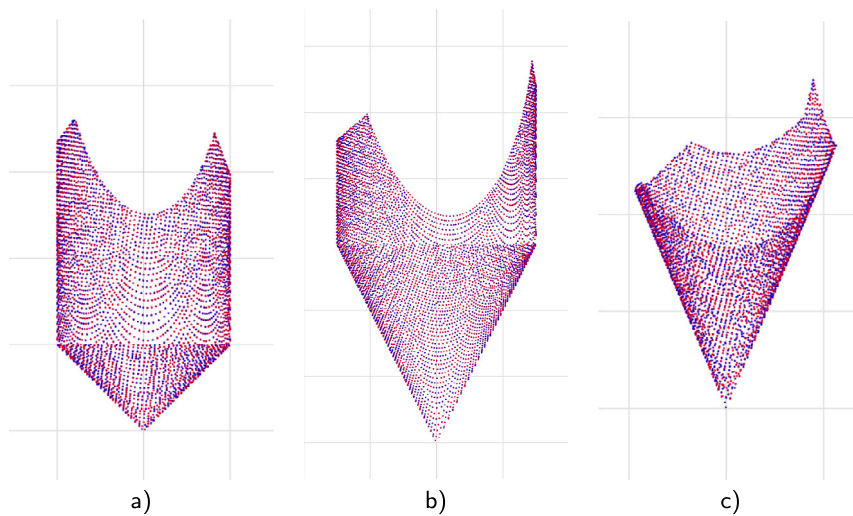

Fig. 10. Perfect synthetic silos (in blue) and their adjustments (in red). (a) $Silo_1$, (b) $Silo_2$ and (c) $Silo_3$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
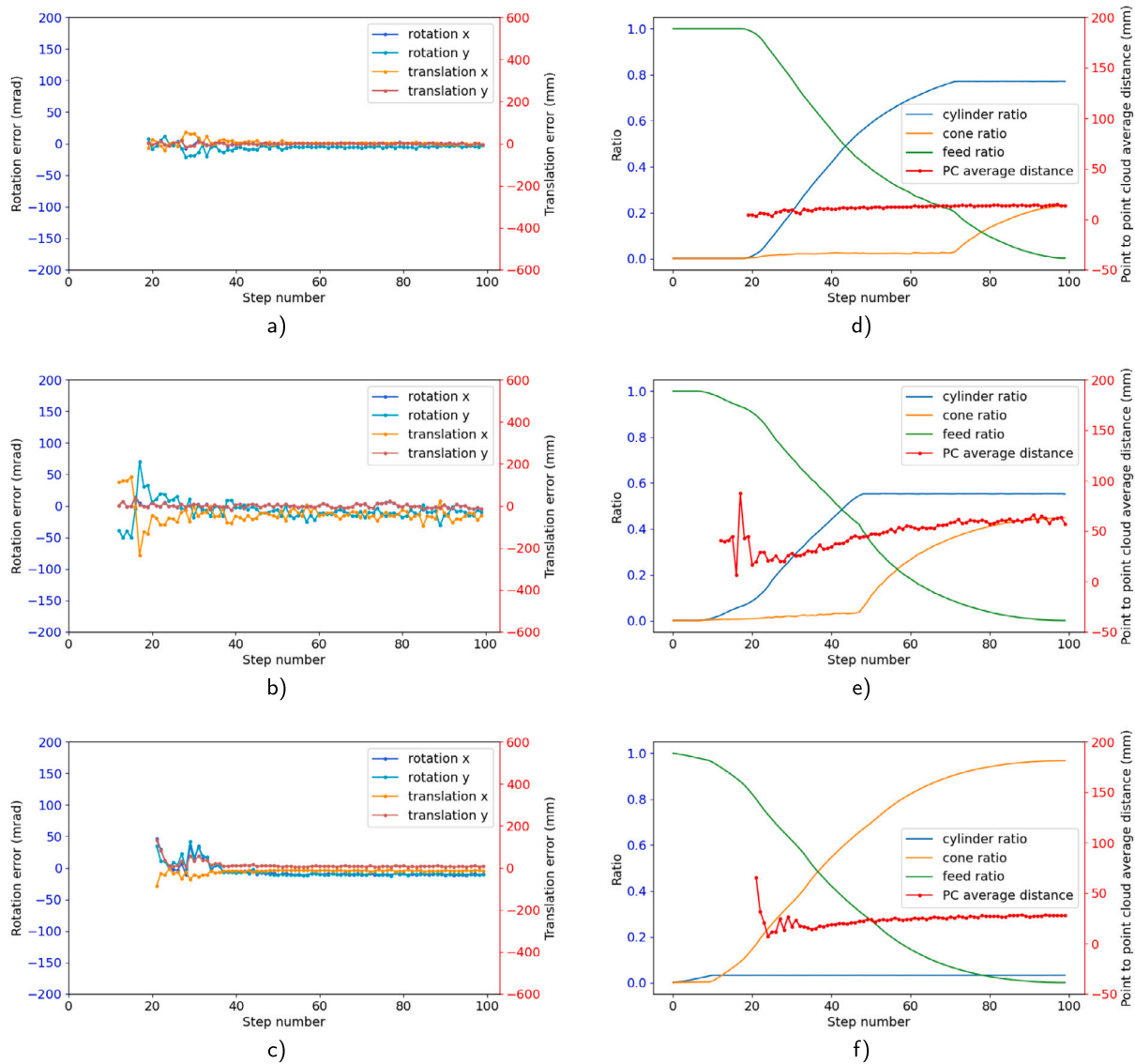
Fig. 11. From (a) to (c), the algorithm output error, i.e., t_x, t_y, r_x, and r_y, at different steps of the algorithm for the perturbed $Silo_1$, $Silo_2$ and $Silo_3$ respectively. From (d) to (e) the cylinder, the cone, and the feed ratios with the PC average distance.

Table 4

For each perturbed synthetic silo, the difference with the real values on t_x and t_y and the PC average distance in mm., and r_x and r_y in mrad. Due to remaining fluctuation the average of the last 10 measures has been used.

Measured error	$Silo_1$	$Silo_2$	$Silo_3$
Translation X (t_x) (mm)	−1.68	−52.29	−12.55
Translation Y (t_y) (mm)	−1.66	−4.50	6.55
Rotation X (r_x) (mrad)	−0.69	−1.56	−10.89
Rotation Y (r_y) (mrad)	−4.21	−10.43	−10.29
PC average distance (mm)	14.06	62.55	27.67

$Silo_2$ has the maximum error since it is the highest one, and the modeled error increases with distance. Concerning $Silo_1$ and $Silo_3$, although $Silo_1$ is higher, the asymmetric perturbations of $Silo_3$ depth maps due to camera position lead to worse adjustments.

Fig. 12 illustrates, for each synthetic silo, the correct adjustment (in blue) and the one obtained with the algorithm estimated parameters (in red). Since perturbed synthetic maps have been used, the output values do not match exactly with the silo characterization of Table 1. The algorithm output adjusts the perturbed silos and not the exact ones.

3.3. Real scenario evaluation

As in the synthetic case, for each real silo, the algorithm outputs (r_x, r_y, t_x, and t_y values) at different discharge steps (see Fig. 13(a–b)) as well as the percentage of points that belongs to the cylinder, the cone or the feed surfaces with the PC average distance (see Fig. 13(c–d)) have been plotted.

In this scenario, the algorithm requires more steps to converge to a stable set of parameters and the convergence values have a higher deviation compared to the synthetic scenario. It happens because the geometry and the feed captured by the camera is not a perfect representation of the reality. The 3D point cloud is slightly distorted because of the camera error and the measurements have noise. Therefore, neither the normal vector estimation and the shape tensors calculation are accurate, mostly because the surfaces are not perfect. These inaccuracies can lead to the inclusion of more feed points into the candidate point selection which can alter the final result. These phenomena has been illustrated in Fig. 14 where the estimated vectors $\widetilde{X}$ before applying PCA to the real (14 (a–c)) and perfect synthetic silos (14(d–f)) are presented. Note how dispersion is much higher on the real cases than the synthetic ones. Focusing on $Silo_{R1}$ (see Fig. 13(a) and (c)), it can be
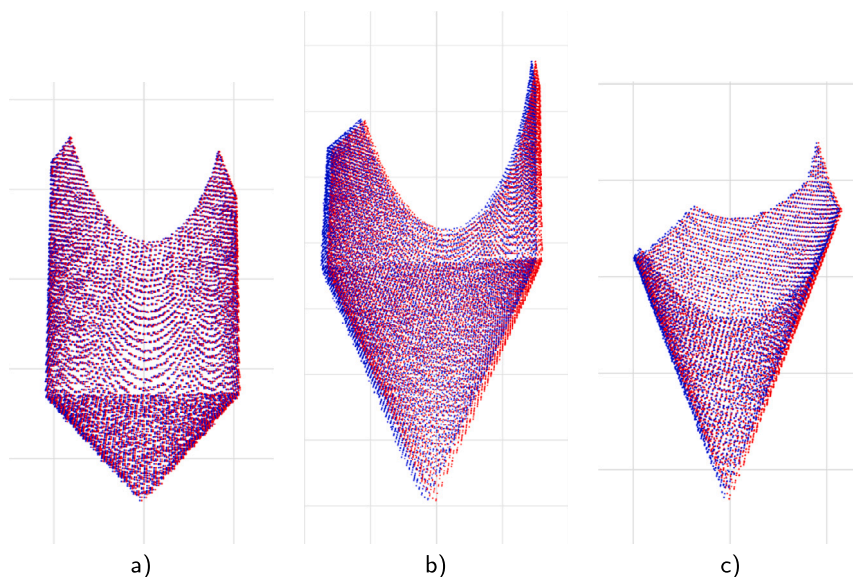
Fig. 12. From (a) to (c), the adjusted resulting perturbed synthetic silos (in red) and the ground truth adjustment (in blue). The adjustment uses the last estimated set of parameters provided by the algorithm. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
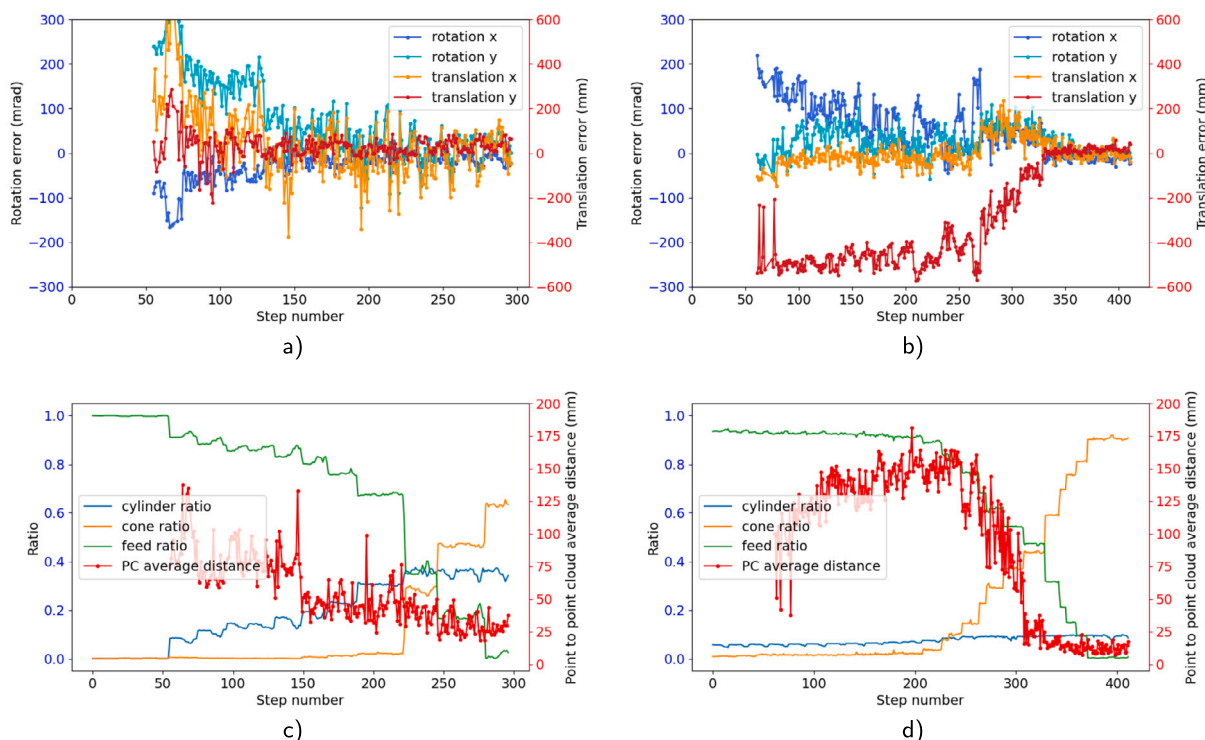

Fig. 13. (a) and (b) the algorithm output error, i.e., t_x, t_y, r_x, and r_y, at different steps of the algorithm for the real $Silo_{R_1}$ and $Silo_{R_2}$ respectively. (c) and (d) the cylinder, the cone, and the feed ratios with the PC average distance. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

seen that the algorithm reached its best accurate parameters after the step 223 when the feed level reached 61% of the total silo height and a structure ratio of 65% from which 46% corresponds to cone points. For $Silo_{R2}$ (see Fig. 13(b) and (d)), the algorithm reached the best accuracy at the step 315 when the feed level reached 61% of the total silo height and a structure ratio of 53% from which 83% corresponds to cone points. Therefore, for real silos the algorithm requires a higher structure ratio to converge and provide stable and accurate results.

In Table 5, for each silo, the errors on r_x, r_y in mrad, and t_x, t_y, and the PC average distance in mm are shown after the results have stabilized. Note that for both silos, the algorithm results have stabilized

at the end of the discharge with a PC average distance of 28.09 mm and 11.94 mm (see PC average distance on the table). Hence, the average adjustment inaccuracy of the algorithm achieved is less than 3 cm of error (see Fig. 15).

3.4. Sensors evaluation

In Table 6, for each tested sensor, the PC averaged distance between the transformed point clouds and the synthetic silo using the measured geometry is shown. Note that there is only a slight difference between
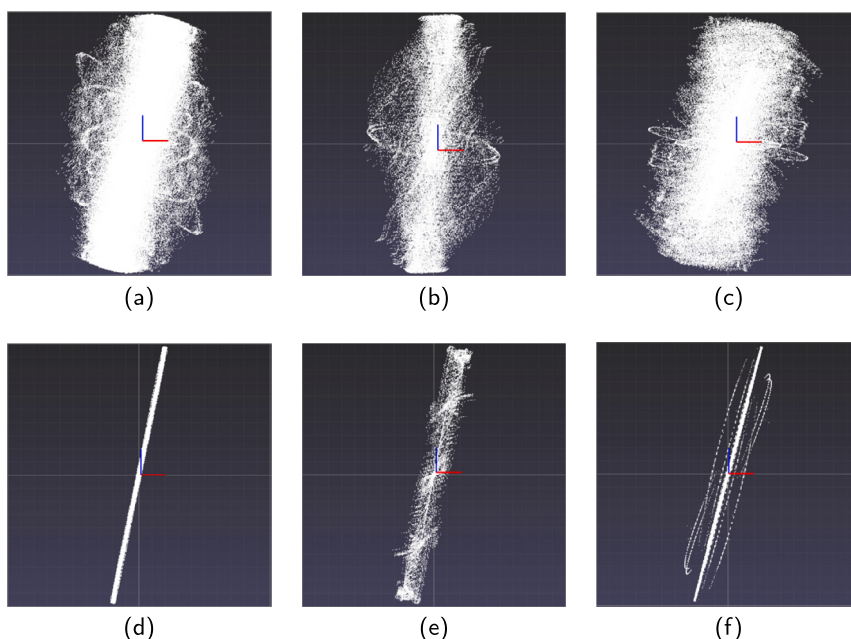
Fig. 14. The collection of estimated vectors $\widetilde{X}$ for real (a–c) and perfect synthetic (d–e) silos.

Table 5
The average error of the last 10 measures after the results stabilized on t_x, t_y and PC average distance (PC average distance) in mm, and the errors on r_X and r_Y in mrad.

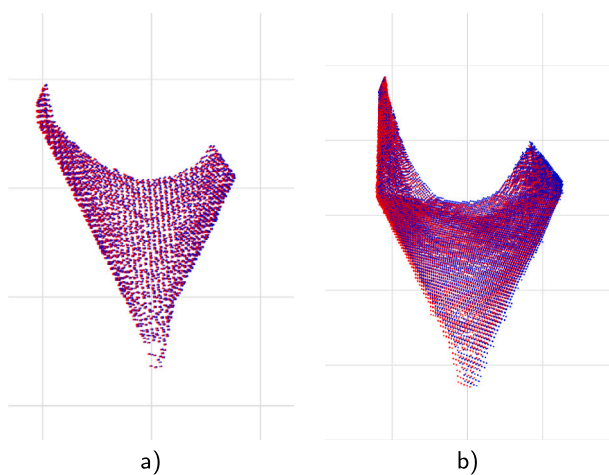| Measured error	$Silo_{R1}$	$Silo_{R2}$
Translation X (t_x) (mm)	−47.92	−38.33
Translation Y (t_y) (mm)	58.50	37.69
Rotation X (r_x) (mrad)	−22.99	−0.02
Rotation Y ($r_x y$) (mrad)	1.01	−0.01
PC average distance (mm)	28.09	11.94

Fig. 15. The ground truth adjustment (blue) compared to the algorithm adjustment result (red) for the real silos. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 16. (a) Captured point clouds in the camera coordinates (Orbbec Astra dark-blue, Intel D415 light-blue, Intel D455 green, Intel L515 yellow) and the well-aligned silo geometry (red). (b) The corresponding adjusted point clouds with the algorithm. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 6
The PC average distance between the different camera acquisitions after being transformed using our method and the silo measured geometry.

	Orbbec Astra	D415	D455	L515
PC average distance (mm)	37.66	59.87	28.06	45.48

cameras which can be due to the errors in the silo geometry measurements. From these results, it can be seen that in the case of small silos the algorithm performance does not depend on the sensor technology. In all situations, the algorithm returns a cylinder axis that adjusts the point cloud to the expected geometry.
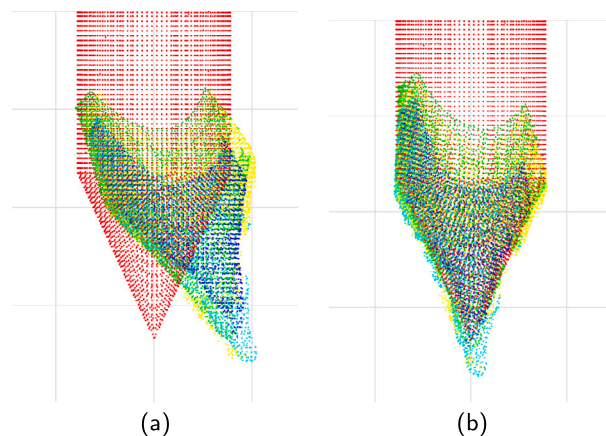
In Table 7, the camera orientation extracted with the accelerometer (Accel values) integrated into the Intel D455 and D515 sensors and the one provided by the proposed algorithm (Proposed method) are presented. Note that the algorithm output is close to the accelerometer values and with similar errors to the ones of the perturbated synthetic scenario (see Table 4). Therefore, the method behaves appropriately regardless of the type of sensor (see Fig. 16).

Table 7
The values of rotation X (rx) and rotation Y(ry) obtained with the accelerometer and the proposed algorithm as well as the PC average distance after applying these rotation values and the PC average distance in mm.

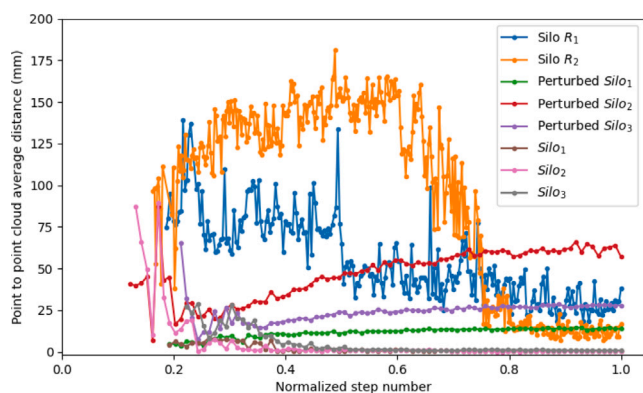|	D455	D515
Accel values rx,ry (mrad)	75.88,312.65	−19.17,305.32
Proposed method rx,ry (mrad)	64.23,307.80	−9.86,300.59
Error rx,ry (mrad)	11.65,4.85	−9.31,4.72
PC average distance (mm)	22.09	18.41

Fig. 17. PC average distance between the estimated adjustment and the ground truth for synthetic and real silos.

3.5. Global evaluation

Experiments have demonstrated the algorithm properly performs in both synthetic and real scenarios as shown in Fig. 17 where the PC average distance of the five analyzed silos with respect to the normalized step number is illustrated. It can be seen that the point cloud average distance between the real and the algorithm adjustments in almost all the cases is under 3 cm except for $Silo_2$ with 6 cm.

Comparing synthetic and real silos results, it has been seen that synthetic silos require less structure to be captured to converge with fewer fluctuations and noise in the results. Therefore, the closer we are to a perfect geometry the more robust the algorithm is. Regarding camera position, it has been seen that better results are obtained when the camera points to the axis and to the cone vertex with rotations in only one of the camera's axis and with a symmetric field of view trying to cover as much structure as possible.

The average running time required to adjust the analyzed silos considering the last discharge image (corresponding to empty silos) is of 8.03 s. Even though the running times are good enough, the implementation has not been optimized and could potentially run faster.

3.6. Limitations

Although the different experiments that have been carried out show the good performance of the proposed approach, there are some limitations that have to be taken into account.

The first one is related to assessing metrics. In our work, we have used the metrics of different structure ratios in order to assess the amount of structure present in each capture. A direct measurement of the surface in the 3D space could have been a more precise metric than measuring the total number of points in a depth map. Depending on the inclination and position of the camera, it could happen that depth maps with the same number of structure points have different equivalent surface structures in the 3D world.

The second one is related to simulated feed shapes. For both the synthetic and real scenarios, we considered very regular feed shapes which differ from silos walls and hence benefit the candidate point

selection and the algorithm performance. For a more complete evaluation, it is necessary to consider complex shapes where the feed can resemble parts of the silo structure in order to evaluate the algorithm's performance in more challenging conditions.

Finally, the size of the silo is another factor that requires further evaluation. In our tests, we have only considered sizes that can be supported by the used devices. For the completeness of the experiments, it will be necessary to consider silos of higher dimensions.

4. Conclusions

In this paper, a new method to support RGB-D sensor-based solutions for the monitoring of silos' content has been proposed. Due to the silo's features, RGB-D sensors can be placed in a limited number of positions in the upper zone of the silo and the acquired data cannot be directly interpreted requiring specific processing. To carry out this processing, a method that only requires a sensor capture, and no information on camera placement nor silo geometry, has been presented. Exploiting the silo's geometry properties, the method takes as input the sensor acquired data and obtains a reference system such that the z-axis coincides with the gravity axis and the xy-plane parallel to the ground plane which is used to obtain the adjusted silo's point cloud. From this, content measures can be easily obtained. The proposed approach has been tested on synthetic and real silos with different geometries, content states, and camera positions, and the obtained results have shown its good performance.

Funding

CRediT authorship contribution statement

Oriol Vila: Conceptualization, Methodology, Software, Validation, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Imma Boada:** Validation, Resources, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration. **Narcis Coll:** Software, Formal analysis, Investigation, Writing – review & editing. **Marta Fort:** Formal analysis, Investigation, Writing – review & editing. **Esteve Farres:** Writing – review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Anon, 2022a. Intel RealSense D415: Available online. URL: https://www.intelrealsense.com/depth-camera-d415.

Anon, 2022b. Orbbec astra: Available online. URL: https://shop.orbbec3d.com/Astra.

Anon, 2022c. Primsense: Available online. URL: http://xtionprolive.com/.

Anon, 2022d. Structure sensor press info. Available online. URL: https://structure.io/.

Cabrera, E., Ortiz, L., da Silva, B., Clua, E., Gonçalves, L., 2018. Versatile method for depth data error estimation in RGB-D sensors. Sensors 18 (9), http://dx.doi.org/10.3390/s18093122.

Camps, J.G., Berenguer, E.F., Sanchez, D.R., Giro, M.A.H., Pla, S.G., 2019. Method and a system for assessing the amount of content stored within a container. In: Google Patents. US Patent 10, 488, 245.

Darwish, W., Tang, S., Li, W., Chen, W., 2017. A new calibration method for commercial RGB-D sensors. Sensors 17 (6), 1204.

Dimitriou, N., Leontaris, L., Vafeiadis, T., Ioannidis, D., Wotherspoon, T., Tinker, G., Tzovaras, D., 2019. Fault diagnosis in microelectronics attachment via deep learning analysis of 3-D laser scans. IEEE Trans. Ind. Electron. 67 (7), 5748–5757.

Dimitriou, N., Leontaris, L., Vafeiadis, T., Ioannidis, D., Wotherspoon, T., Tinker, G., Tzovaras, D., 2020. A deep learning framework for simulation and defect prediction applied in microelectronics. Simul. Model. Pract. Theory 100, 102063.

do Carmo, M.P., 1976. Differential Geometry of Curves and Surfaces. Prentice-Hall, Englewood Cliffs, New Jersey.

Fu, L., Gao, F., Wu, J., Li, R., Karkee, M., Zhang, Q., 2020. Application of consumer RGB-D cameras for fruit detection and localization in field: A critical review. Comput. Electron. Agric. 177, 105687. http://dx.doi.org/10.1016/j.compag.2020.105687, URL: https://www.sciencedirect.com/science/article/pii/S0168169920319530.

Fu, M.C., Hu, J.Q., 2012. Conditional Monte Carlo: Gradient Estimation and Optimization Applications, Vol. 392. Springer Science & Business Media.

Giancola, S., Valenti, M., Sala, R., 2018. A Survey on 3D Cameras: Metrological Comparison of Time-of-Flight, Structured-Light and Active Stereoscopy Technologies. Springer.

Goldfeather, J., Interrante, V., 2004. A novel cubic-order algorithm for approximating principal direction vectors. ACM Trans. Graph. 23 (1), 45–63. http://dx.doi.org/10.1145/966131.966134.

Han, J., Shao, L., Xu, D., Shotton, J., 2013. Enhanced computer vision with microsoft kinect sensor: A review. IEEE Trans. Cybern. 43 (5), 1318–1334.

Holz, D., Ichim, A.E., Tombari, F., Rusu, R.B., Behnke, S., 2015. Registration with the point cloud library: A modular framework for aligning in 3-d. IEEE Robot. Autom. Mag. 22 (4), 110–124.

Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W., 1992. Surface reconstruction from unorganized points. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '92, Association for Computing Machinery, New York, NY, USA, pp. 71–78. http://dx.doi.org/10.1145/133994.134011.

Idoje, G., Dagiuklas, T., Iqbal, M., 2021. Survey for smart farming technologies: Challenges and issues. Comput. Electr. Eng. 92, 107104. http://dx.doi.org/10.1016/j.compeleceng.2021.107104, URL: https://www.sciencedirect.com/science/article/pii/S0045790621001117.

Ingman, M., Virtanen, J.P., Vaaja, M.T., Hyyppä, H., 2020. A comparison of low-cost sensor systems in automatic cloud-based indoor 3D modeling. Remote Sens. 12 (16), 2624.

Kehl, W., Milletari, F., Tombari, F., Ilic, S., Navab, N., 2016. Deep learning of local RGB-D patches for 3d object detection and 6d pose estimation. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14. Springer, pp. 205–220.

Kolhatkar, C., Wagle, K., 2020. Review of SLAM algorithms for indoor mobile robot with LIDAR and RGB-D camera technology. Innov. Electr. Electron. Eng. 397–409.

Lewis, J., 2004. Technology Review Level Measurement of Bulk Solids in Bins, Silos and Hoppers. Monitor Technologies LLC.

Liu, H., Qu, D., Xu, F., Zou, F., Song, J., Jia, K., 2020. Approach for accurate calibration of RGB-D cameras using spheres. Opt. Express 28 (13), 19058–19073.

Liu, R., Shen, J., Chen, C., Yang, J., 2019. SLAM for robotic navigation by fusing RGB-D and inertial data in recurrent and convolutional neural networks. In: 2019 IEEE 5th International Conference on Mechatronics System and Robots. ICMSR, IEEE, pp. 1–6.

Park, K.B., Choi, S.H., Kim, M., Lee, J.Y., 2020. Deep learning-based mobile augmented reality for task assistance using 3D spatial mapping and snapshot-based RGB-D data. Comput. Ind. Eng. 106585.

Raba, D., Gurt, S., Vila, O., Farres, E., et al., 2020. An Internet of Things (IoT) solution to optimise the livestock feed supply chain. In: CS & IT Conference Proceedings, Vol. 10, No. 4. CS & IT Conference Proceedings.

Raba Sánchez, D., 2021. Applications of the Internet of Things and optimization to inventory and distribution management (Ph.D. thesis). Universitat Oberta de Catalunya (UOC).

Rousopoulou, V., Papachristou, K., Dimitriou, N., Drosou, A., Tzovaras, D., 2019. Automated mechanical multi-sensorial scanning. In: International Conference on Computer Vision Systems. Springer, pp. 433–442.

Sun, Q., Yuan, J., Zhang, X., Duan, F., 2020. Plane-Edge-SLAM: Seamless fusion of planes and edges for SLAM in indoor environments. IEEE Trans. Autom. Sci. Eng.

Tang, Y., Dananjayan, S., Hou, C., Guo, Q., Luo, S., He, Y., 2021. A survey on the 5G network and its impact on agriculture: Challenges and opportunities. Comput. Electron. Agric. 180, 105895. http://dx.doi.org/10.1016/j.compag.2020.105895, URL: https://www.sciencedirect.com/science/article/pii/S0168169920331008.

Tao, W., Zhao, L., Wang, G., Liang, R., 2021. Review of the internet of things communication technologies in smart agriculture and challenges. Comput. Electron. Agric. 189, 106352. http://dx.doi.org/10.1016/j.compag.2021.106352, URL: https://www.sciencedirect.com/science/article/pii/S0168169921003690.

Zhang, Z., 2012. Microsoft kinect sensor and its effect. IEEE Multimed. 19 (2), 4–10.