Automatika

Journal for Control, Measurement, Electronics, Computing and Communications

ISSN: 0005-1144 (Print) 1848-3380 (Online) Journal homepage: https://www.tandfonline.com/loi/taut20

Visual servoing for low-cost SCARA robots using an RGB-D camera as the only sensor

P. Đurović, R. Grbić & R. Cupec

To cite this article: P. Đurović, R. Grbić & R. Cupec (2017) Visual servoing for low-cost SCARA robots using an RGB-D camera as the only sensor, Automatika, 58:4, 495-505, DOI: 10.1080/00051144.2018.1461771

To link to this article: https://doi.org/10.1080/00051144.2018.1461771

AUTOMATIKA, 2017
VOL. 58, NO. 4, 495–505
https://doi.org/10.1080/00051144.2018.1461771

REGULAR PAPER

OPEN ACCESS Check for updates

Visual servoing for low-cost SCARA robots using an RGB-D camera as the only sensor

P. Đurović, R. Grbić and R. Cupec

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Osijek Croatia

ABSTRACT

Visual servoing with a simple, two-step hand–eye calibration for robot arms in Selective Compliance Assembly Robot Arm configuration, along with the method for simple vision-based grasp planning, is proposed. The proposed approach is designed for low-cost, vision-guided robots, where tool positioning is achieved by visual servoing using marker tracking and depth information provided by an RGB-D camera, without encoders or any other sensors. The calibration is based on identification of the dominant horizontal plane in the camera field of view, and an assumption that all robot axes are perpendicular to the identified plane. Along with the plane parameters, one rotational movement of the shoulder joint provides sufficient information for visual servoing. The grasp planning is based on bounding boxes of simple objects detected in the RGB-D image, which provide sufficient information for robot tool positioning, gripper orientation and opening width. The developed methods are experimentally tested using a real robot arm. The accuracy of the proposed approach is analysed by measuring the positioning accuracy as well as by performing grasping experiments.

ARTICLE HISTORY
Received 14 November 2017
Accepted 29 March 2018

KEYWORDS
Visual servoing; hand–eye calibration; grasp planning; SCARA; low-cost robot arm

1. Introduction

Although robot sales increasingly grow every year, robots are today mostly used in the industry [1]. The goal of making robots more affordable to a wide community of developers as well as for everyday applications motivated a number of research teams to design low-cost robotic solutions [2–4]. The research presented in this paper contributes to this goal by proposing a method for vision-based control of a particular type of low-cost robot arms. The presented work comprehends hand–eye calibration for the purpose of visual servoing and grasping of simple convex objects.

In this paper, low-cost robot manipulators based on stepper motors which do not have absolute encoders or any other proprioceptive sensors for measuring joint angles and rely on visual feedback only are considered. Lack of absolute encoders implies that tool positioning cannot be performed by inverse kinematics, since absolute joint angles cannot be known or preset. Therefore, in this paper, an approach for relative tool positioning based on computer vision is proposed. The target position of the tool is determined by detecting objects of interest in an image of the robot's workspace acquired by an RGB-D camera, while the current tool position is obtained by localization of a marker mounted on the robot arm close to the end effector using a vision-based marker tracking software. The visual servoing method, proposed in this paper, computes changes of joint angles required to move the tool from its current position to the target position. The method is designed for robots in Selective Compliance Assembly Robot Arm (SCARA) configuration, which is common in robot manipulation since it is advantageous for planar tasks [5], such as assembly or pick-and-place. In the particular configuration, the current and the target tool position can be represented by points on two circles of different radii centred in the shoulder joint axis. The distance between the tool and the shoulder joint axis is adjusted by changing the elbow joint angle, while the shoulder joint moves the tool along the circle until the target position is reached. Considering imperfection of the robot motors as well as the uncertainty of the information about the pose of the shoulder axis w.r.t. the camera, the tool position is corrected iteratively until the given position is reached within a specified tolerance, which is validated by the robot's vision system.

The pose of the shoulder joint axis w.r.t. the camera, required by the proposed visual servoing approach, is achieved by a novel, simple and fast, two-step hand–eye calibration. Since the proposed visual servoing approach is based on the tool distance w.r.t the shoulder joint axis and its position on the circle centred in this axis, it can be applied only in the eye-to-hand configuration, where the camera is mounted in a fixed position w.r.t. the shoulder joint axis. The eye-to-hand

CONTACT P. Đurović pdurovic@etfos.hr Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Osijek Croatia

configuration is suitable for small robot arms, where the camera couldn't be mounted on the robot's end effector. Furthermore, the eye-to-hand configuration is typical for anthropomorphic robots, as well as for biological systems, i.e. humans and animals, where the vision system positioned high above the ground provides a wider overview of the environment. For the same reason, this configuration is suitable for mobile robot manipulators, where the same camera can be used for robot localization, search for the object of interest in the robot's environment and manipulation with objects.

We consider the SCARA configuration, where all joint axes are parallel to the gravity axis, and assume that the objects of interest are positioned on a horizontal plane, referred to in this paper as the *supporting plane*. The proposed hand–eye calibration method identifies the dominant plane in the camera field of view and determines the shoulder joint axis orientation as the vector perpendicular to this plane. Besides the supporting plane information, the proposed calibration method requires only one rotational movement by the shoulder joint. Assuming that the elbow joint remains still, a shoulder joint movement slides the tool along a circle centred in the shoulder joint axis. By knowing the change in the shoulder joint angle, the centre of this circle w.r.t. the camera can be determined. The shoulder joint axis is determined as the line perpendicular to the supporting plane, which passes through the circle centre. The simplicity of the proposed method makes it suitable for often recalibration.

In order to obtain a complete vision-based tool positioning system for SCARA robots, we developed a simple method for grasp planning, where the target tool position is computed based on visual input. The proposed grasp planning approach consists in detecting simple convex objects in an RGB-D image of the scene, creating the object bounding boxes and computing the target tool position, orientation and opening width of the gripper based on the bounding box of one of these objects. The tool orientation is determined according to the shape and orientation of the object of interest. More precisely, the region of the object's surface of the smallest curvature, approximately perpendicular to the supporting plane, is considered to be suitable position for the contact points of the gripper fingers. Low curvature regions on the object's surface are detected by segmenting this surface into approximately planar patches. The grasp planning method proposed in this paper, however, has some limitations: (i) it is assumed that the tool has only one rotational DoF and that objects can be grasped only from above, (ii) grasping is possible only for convex objects and (iii) stable grasp is not guaranteed even for convex objects. Nevertheless, the class of objects to which the proposed method can be applied is still wide and the efficiency of this method is clearly its advantage in comparison to more general, but also more complex approaches, some of which are reviewed in Section 2.

There are two contributions of this paper. The first contribution is a novel visual servoing approach for SCARA robots without absolute encoders and with a camera as the only sensor, which uses the information about the shoulder joint axis obtained by a simple hand–eye calibration method. Another contribution is a method for grasp planning, suitable for simple convex objects detected in RGB-D images. The proposed methods are experimentally evaluated on a set of visual servoing and grasping experiments. These experiments are performed using a low-cost, vision-based robot arm in SCARA configuration.

The paper is structured as follows. In Section 2, we present an overview of the current state of the art in the fields of low-cost robotics, visual servoing, hand–eye calibration and grasp planning. The proposed visual servoing approach with the associated calibration method is explained in Section 3. In Section 4, the grasp planning method based on visual input is proposed. Finally, an experimental evaluation of the proposed approaches is presented in Section 5. The last section brings a conclusion and options for future research.

2. Related research

This section provides a review of published research closely related to the work presented in this paper. The reviewed research is from the fields of low-cost robots, hand–eye calibration, visual servoing and vision-based grasp planning.

Low-cost vision-guided robot manipulators: Design of low-cost robots has been a topic of interest of several research groups [2–4]. In [2], a counterbalance mechanism, which reduces cost, is proposed. The presented setup, however, doesn't include vision. A low-cost, custom-made, 6 DoF Pieper-type robotic manipulator with eye-in-hand configuration is proposed in [3]. In [4], a vision-based robot system consisting of an off-the-shelf 4 DoF robot arm and a camera is presented. This system is based on visual servoing without encoders, which uses a hand–eye calibration method requiring two movements of the robot arm.

Hand–eye calibration and visual servoing: In [3,6–14] calibration methods for the eye-in-hand configuration are proposed. In [8], an eye-in-hand calibration method is proposed, consisting of a small number of steps, where a light plane projected by a laser on the end effector is being used in calibration. A study which performs eye-in-hand calibration and intrinsic camera calibration at the same time is proposed in [9]. This method relies on a single point, which must be visible during the calibration. The point is not placed on the robot itself, but in the robot's environment. In [10], a combined

internal camera parameter and eye-to-hand calibration approach, which uses a calibration panel, is proposed. This approach uses A4 size printed checkerboard as the calibration panel, which is mounted on the robot end effector. However, this method isn't suitable for smaller robot arms and automatic recalibration, because of the size of the calibration panel. A more complex calibration, using global polynomial optimization, is given in [11], and it uses eye-in-hand camera configuration. In [3], visual servoing is implemented on a custom developed robot arm. In this setup, an eye-in-hand configuration is considered, where the pose of the target is extracted from a 2D image by photogrammetry. In [4], a visual servoing for absolute positioning is presented. The hand–eye calibration method consists of two rotational movements forming a triangle of points sufficient for computing the pose of the robot reference frame (RF) w.r.t. the camera. However, due to the small number of measurements used in this computation, a high accuracy cannot be achieved. Furthermore, this method requires the robot to be positioned in an appropriate initial position before performing the hand–eye calibration, and it is not clear how this could be done automatically.

Grasp planning: In a number of researches, the position of the grasped object is extracted from a 2D image based on the object contours [15,16]. In order to avoid computations in 3D, computations are sometimes transformed into 2D space [17]. In [18], along with a stereo camera, laser scanners are used to obtain the 3D position of objects in the scene. Information about 3D geometry of the scene is clearly useful in robotic manipulation, which is being demonstrated in the results of recent computer vision research [19–24]. In order to achieve a real-time performance, a database of graspable objects is created in [25], which allows off-line determination of a successful grasp for a certain object. The graspable objects are represented by CAD models or complete 3D scans. The authors of the work presented in [26] implemented the grasp planning for the non-convex objects by the object decomposition within the motion planner tool Move3D [27]. A method which computes grasps based on bounding boxes is given in [17]. A set of graspable bounding boxes with different properties, such as size and orientation, is computed off-line and stored in a database. For a given bounding box of an object detected in the scene, a similar bounding box is found in the database and, if possible, grasping is performed with an off-line computed gripper configuration. The off-line computation time needed was nearly nine minutes. The approach to grasp planning used in [28] is to determine the gripper configuration by optimization. In the case of complex grasp planning tasks, the grasp proposals are evaluated by simulators. Among the other grasping simulators, Graspit! [29], being one of the most famous, is used by Xue and Dillmann [17], Marton et al. [18] and Kragic et al. [25]. In [25], Graspit! is used to integrate real-time vision with online grasp planning, where it executes stable grasp on objects and monitors the grasped object's trajectory as it moves.

3. Tool positioning using an RGB-D camera

This section describes the robot tool positioning approach based on visual servoing with a belonging novel hand–eye calibration method. The discussed approach consists of determining the current tool position and the target position w.r.t. the robot shoulder joint axis using computer vision and changing the joint angles until these two positions match within a user-specified tolerance.

3.1. Measuring the marker position using an RGB-D camera

The developed methods are designed for a robot manipulator with a marker placed on the robot arm. The centre of the marker is used to determine the current position of the robot end effector. Since an RGB-D camera is used in the considered robot system, two measurements of the marker distance w.r.t. the camera are available: the one obtained by the RGB image using a marker tracking software and the other obtained by the depth sensor. The marker tracking software detects the marker in the RGB image and computes its position w.r.t. the camera RF according to the size of the marker in the image and the known actual marker size specified by the user. This position is defined by coordinates $(x_{\mathrm{RGB}}, y_{\mathrm{RGB}}, z_{\mathrm{RGB}})$. The depth sensor of the RGB-D camera assigns depth values to the image points. The depth value of an image point is its z-coordinate w.r.t. the camera RF, where the z-axis of this RF is parallel to the camera's optical axis. The depth value z_d assigned to the image point representing the marker centre is an alternative measurement of the z-coordinate of the marker centre. We perform fusion of the two measurements of the marker centre z-coordinate, z_{RGB} and z_d, by taking into consideration the uncertainty of both measurements. In order to estimate the uncertainty of the market tracking software, the marker is moved along a vertical line by controlling the translational joint of the robot. The orthogonal distance regression (ODR) line is fitted to the set of points representing the marker centre positions obtained by the marker tracking software. For each of these marker centre positions, the difference between its z-coordinate and the z-coordinate of the point on the optical ray passing through the marker centre closest to the ODR line is computed. This difference is regarded as the error in measurement of z_{RGB}. The variance of this error, representing a measure of the uncertainty of z_{RGB}, is denoted by σ^2_{RGB}. In order to estimate the uncertainty of z_d, a planar surface

is placed in the camera field of view, the points belonging to this surface are identified in the depth image and the uncertainty of z_d is estimated by a statistical analysis of the deviations of these points from the least-squares plane fitted to them. The discussed planar surface is identified using the standard random sample consensus (RANSAC)-approach [30]. Each point belonging to this surface is projected onto the least-squares plane along the optical ray passing through this point. The difference between the z-coordinates of a particular point and its projection is regarded as the measurement error. The variance of this measurement error σ_d^2 is used as a measure of uncertainty of z_d. Given two measurements, z_{RGB} and z_d and their variances σ_{RGB}^2 and σ_d^2, the optimal z-coordinate is computed by

$$z = \frac{\frac{z_d}{\sigma_d^2} + \frac{z_{\text{RGB}}}{\sigma_{\text{RGB}}^2}}{\frac{1}{\sigma_d^2} + \frac{1}{\sigma_{\text{RGB}}^2}}. \tag{1}$$

This z-coordinate is used to correct the other two coordinates of the marker centre by scaling them using the scaling factor

$$s = \frac{z}{z_{\text{RGB}}}. \tag{2}$$

Assuming that the marker is clearly visible, the described method provides the 3D position of the marker w.r.t. the camera RF denoted by ${}^C p_M$. Computation of the marker position is given in Appendix 1.

3.2. Visual servoing

Before explaining the developed approaches, the notation used in the rest of this paper is introduced. In this paper, ${}^B t_A$ denotes the translation vector defining the position of an RF S_A w.r.t. an RF S_B. Furthermore, the following notation is used for RFs: R represents robot, C camera, M marker RF and G represents RF of the object bounding box. The position of a point A w.r.t. the RF B is denoted by ${}^B p_A$.

There are two common variants of the SCARA configuration: the one where the first joint is translational and the other two are rotational, and the other with the first two rotational joints, and the third translational joint. In this paper, it is assumed that the first joint is translational, but the same method is applicable to both of these two configurations.

The purpose of visual servoing is to compute the changes of joint variables required to move the marker attached to the robot arm from its current position to a given target position. The current marker position is measured using the approach described in Section 3.1, while the target position is determined by the grasp planning method presented in Section 4. The proposed visual servoing approach requires the pose of the shoulder joint axis w.r.t. the camera RF, defined by unit vector ${}^C z_R$, representing the axis orientation, and vector ${}^C t_R$,

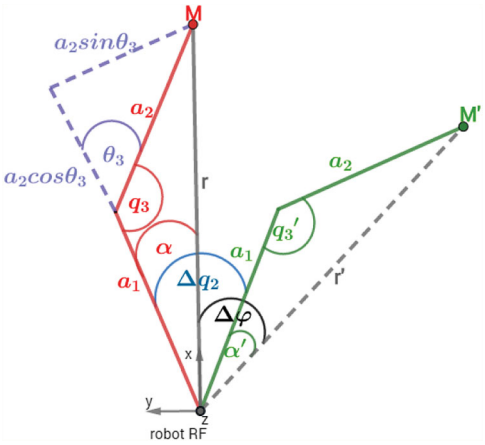

Figure 1. Robot arm in the current (M) and target (M′) position.

representing the position of a reference point on the considered axis w.r.t. the camera RF. Vectors ${}^C z_R$ and ${}^C t_R$ are obtained by the hand–eye calibration described in Section 3.3. Visual servoing can be performed multiple times with the same parameters ${}^C z_R$ and ${}^C t_R$ as long as there is no need for recalibration.

Let us consider the SCARA configuration geometry shown in Figure 1, where a_1 and a_2 represent the lengths of robot arm links. In order to explain the considered visual servoing approach, we introduce the robot RF centred in a reference point of the shoulder joint axis, with z-axis identical to the shoulder joint axis. We define the other two axes of the robot RF according to the current marker position. The x-axis is directed towards the marker, as shown in Figure 1. In each correction step of the visual servoing, x- and y-axes of the robot RF are redefined.

Let M be the current marker position and M' the target marker position. The visual servoing computes the required change in vertical position Δz, which is achieved by the first translational joint, as well as the required changes Δq_2 and Δq_3 of the second and the third rotational joint. The changes in the rotational joints are computed based on the geometry shown in Figure 1.

The required change of the translational joint variable, Δz, represents the difference in the z-coordinate of the robot RF and is independent of the other joint variables. To compute Δz, only the current coordinate z and the target coordinate z' of the marker w.r.t. the robot RF are required. Therefore, Δz is computed by

$$\Delta z = z' - z, \tag{3}$$

where

$$z = {}^C z_R^\top ({}^C p_M - {}^C t_R) \tag{4}$$

and z' is computed analogously.

The required change in the elbow joint Δq_3 is computed by

$$\Delta q_3 = q_3' - q_3, \tag{5}$$

where q_3 represents the current angle of the elbow joint, and it is calculated as in the standard planar robot manipulator configuration [31]. Analogously, q_3' represents the angle of this joint in the target position.

The required change in the shoulder joint Δq_2 is computed by

$$\Delta q_2 = \alpha - \alpha' + \Delta\varphi, \tag{6}$$

where $\Delta\varphi$ represents the angle between vectors r and r', shown in Figure 1, while α is computed by

$$\alpha = a\sin\frac{a_2 \sin q_3}{\|r\|} \tag{7}$$

and α' is computed analogously.

Vector r, connecting the shoulder joint axis and the current marker position, is computed by

$$r = {}^C p_M - {}^C t_R - z \cdot {}^C z_R, \tag{8}$$

and vector r', connecting the shoulder joint axis and the target marker position, is computed analogously to r.

This algorithm is repeated iteratively until the marker reaches the target position, within a given tolerance. This tolerance represents the maximal acceptable distance between the target and the obtained position. It shouldn't be zero because of the measurement noise, backlash and limited robot precision, which prevent the robot to achieve the exact target position and could cause the visual servoing to end in an infinite loop.

The positioning accuracy of the considered robot system depends on the accuracy of the marker position measured by vision, and on the accuracy of the shoulder joint axis orientation w.r.t. the camera estimated by detection of the supporting plane, as explained in Section 3.3. The accuracy of determining the reference point ${}^C t_R$ doesn't impact the positioning accuracy, but impacts the number of visual servoing iterations. A more accurate estimation of ${}^C t_R$ results in fewer iterations.

3.3. Hand–eye calibration for relative positioning

Parameters of the shoulder joint axis, ${}^C z_R$ and ${}^C t_R$, required for the visual servoing algorithm described in Section 3.2, are determined by the calibration procedure described in this section. The calibration method proposed in this paper determines the shoulder joint axis orientation by detecting the supporting plane. The position of this axis is defined by a reference point, which is an arbitrary point on this axis determined by performing only one rotational movement of the shoulder joint. The supporting plane is estimated using the RANSAC algorithm explained next. First, three random points from the RGB-D image are selected and parameters of the plane passing through those points are computed. All points belonging to that plane, within a given threshold, represent the consensus set. This procedure is repeated for a given number of times and the parameters of the plane with the greatest consensus set are selected. Finally, the least-square plane is fitted to the selected consensus set. An example of the determined supporting plane is given in Figure 2. Orientation of the determined supporting plane normal concurs with the shoulder joint axis orientation ${}^C z_R$.

Figure 2. Detection of the dominant plane.

Now, let us consider the robot movement, where only the shoulder joint is being rotated for a known angle of rotation Δq_2 causing the marker to move from the initial point $M(0)$ to the final point $M(1)$. Rotation of point M about an axis passing through a point defined by vector ${}^C t_R$, where the axis orientation is defined by vector ${}^C z_R$, can be described by equation

$$R({}^C z_R, \Delta q_2) \cdot ({}^C p_{M(0)} - {}^C t_R) = {}^C p_{M(1)} - {}^C t_R, \tag{9}$$

where $R({}^C z_R, \Delta q_2)$ denotes the rotation matrix defined by vector ${}^C z_R$ and angle Δq_2. Vector ${}^C t_R$ can be computed by solving Equation (9) for ${}^C p_{M(0)}$ and ${}^C p_{M(1)}$ obtained by the marker tracking software. The proposed hand–eye calibration method consists of the steps explained in Appendix 2.

4. Vision-based simple objects grasp planning

In this section, an approach for grasp planning based on bounding boxes of objects detected in RGB-D images is presented. It is assumed that the gripper is capable for grasping objects from above only, which is typical for SCARA robots. In order to facilitate successful grasping, a suitable orientation of the gripper, its position above the object and the gripper opening width are required. The proposed grasp planning approach is limited to simple convex objects. Objects of interest are detected in an RGB-D image of the robot's workspace using the method presented in [32]. Basically, the RGB-D image is segmented into planar patches and adjacent planar patches are aggregated into objects using a criterion based on convexity. Hence, the result of this method is one or multiple objects, each represented by a set of planar patches. Considering only grasping from

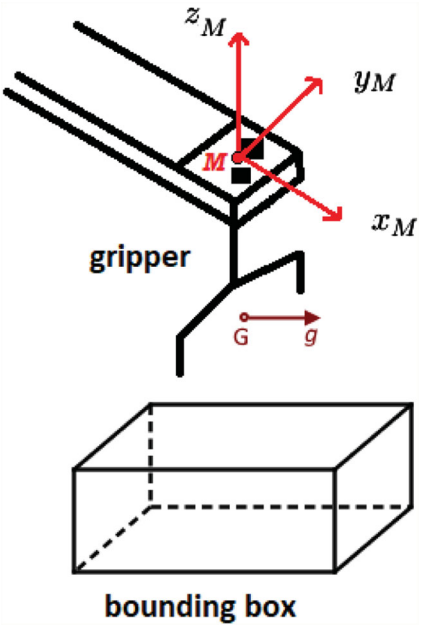

Figure 3. Marker RF and tool orientation.

above and assuming that the objects lie on the supporting plane, it is assumed that a low curvature object surface, oriented at a steep angle w.r.t. the supporting plane, provides a stable grasping point. Grasp vector g, which defines the gripper orientation, as shown in Figure 3, is perpendicular to the supporting plane and the normal of one of the objects planar patches. Hence, it can be computed by

$$g = \frac{n_s \times n_i}{\|n_s \times n_i\|}, \tag{10}$$

where n_s represents the normal of the supporting plane, and n_i represents the normal of the ith planar patch of the grasped object. The planar patch used for computing vector g is chosen in such a way that g computed by Equation (10) has the minimum orientation uncertainty.

The uncertainty of g depends on the uncertainty of n_i as well as on its orientation. The uncertainty of n_i is estimated using the approach described in [33]. Covariance matrix Σ_p of all points belonging to the considered patch is computed. The eigenvector corresponding to the smallest eigenvalue of Σ_p represents the planar patch normal, while the other two eigenvalues describe the distribution of points in the planar patch plane. Those two values are greater for larger planar patches corresponding to low-curvature regions of the object's surface. The planar patch normal uncertainty is represented by the following equation:

$$n_i = \hat{n}_i + M_i s_i, \tag{11}$$

where n_i is the true value of the planar patch normal, $\hat{n}_i$ is its measured value, M is the matrix whose diagonal elements are the eigenvectors corresponding to two largest eignevalues of Σ_p and s_i is a disturbance vector representing the deviation of n_i from $\hat{n}_i$ in

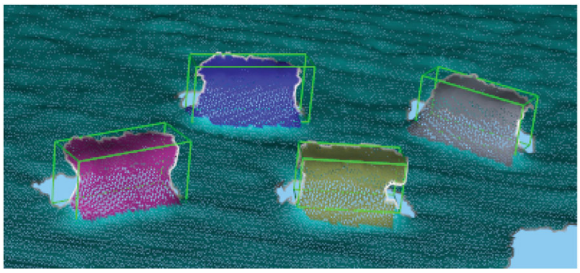

Figure 4. An example of bounding boxes of objects detected in the scene.

two directions perpendicular to $\hat{n}_i$. Covariance matrix Σ_{n_i}, which represents the distribution of the disturbance vector s_i, is a diagonal matrix whose diagonal elements are approximately inversely proportional to the two greater eigenvalues of Σ_p [33]. Hence, the larger planar patches have smaller normal uncertainty. The uncertainty of vector g can be estimated by propagating the uncertainty of n_i. The covariance matrix Σ_g, representing the uncertainty of g, is computed by

$$\Sigma_g = \frac{\mathrm{d}g}{\mathrm{d}s_i} \cdot \Sigma_{n_i} \cdot \left(\frac{\mathrm{d}g}{\mathrm{d}s_i}\right)^{\top}, \tag{12}$$

where $\mathrm{d}g/\mathrm{d}s_i$ represents a Jacobian, computed by substituting Equation (11) into Equation (10) and partially deriving the obtained vector w.r.t. the components of s_i. Finally, the measure of orientation uncertainty of vector g is computed as the projection of Σ_g in the direction perpendicular to g. This projection is computed by

$$\sigma_g = u^{\top} \Sigma_g u, \tag{13}$$

where u represents the unit vector perpendicular to both g and n_s. Value σ_g is computed for every planar patch of an object and vector g is computed using the normal of the planar patch corresponding to the smallest σ_g.

A stable grasp is determined by computing the bounding box of the considered object, whose sides are aligned with vectors n_s, g and u. Examples of objects detected in an RGB-D image of the robot's workspace and their bounding boxes are shown in Figure 4.

The basic idea of our approach comes from the fact that if the line connecting the grasping points passes through the object centre of gravity and if it is approximately perpendicular to the surface normals in the grasping points, the grasp will be stable. We assume that the object centre of gravity is close to its bounding box centre of gravity. Therefore, the grasping points are defined in such a way that the connection line between the grasping points passes through the bounding box centroid. The bounding box plane parameters provide sufficient information for computing the object centroid, a point ${}^{C}p_G$, representing the target position for visual servoing, i.e. the midpoint between the two gripper fingers.

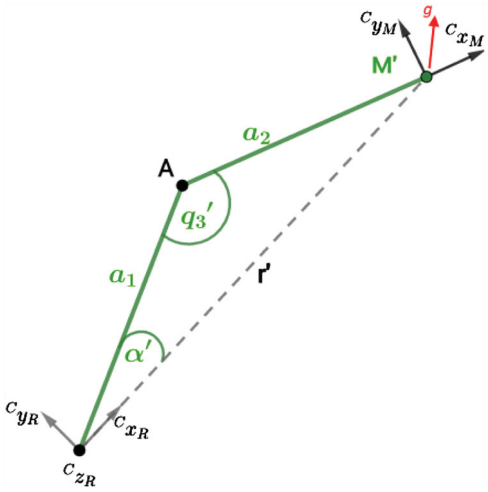

Figure 5. Robot and end effector RFs in position $^{C}p_{M'}$.

After positioning of the robot arm above the target point, rotation is performed by the angle of rotation, q_4, which represents the angle between $^{C}x_M$ and g, as shown in Figure 5. Vectors $^{C}x_M$ and $^{C}y_M$ represent the axes of the marker RF, as shown in Figures 3 and 5. Vector $^{C}x_M$ is parallel to the second link of the robot arm, denoted in Figure 5 by a_2. It is computed as the unit vector parallel to the line connecting the marker centre M and point A on the third joint axis. Vector $^{C}y_M$ is perpendicular to $^{C}z_R$ and $^{C}x_M$. The position of point A w.r.t. the camera RF is computed by

$$^{C}p_A = {}^{C}t_R + a_1 \cdot ({}^{C}x_R \cdot \cos\alpha' + {}^{C}y_R \cdot \sin\alpha') - z' \cdot {}^{C}z_R, \tag{14}$$

where α' and z' are explained in Section 3.2.

Finally, the gripper opening width is computed as the distance between the two bounding box faces parallel to the vector g.

5. Experimental evaluation

In this section, an experimental analysis of the proposed methods for visual servoing and grasp planning is presented.

5.1. Experimental setup

The robot manipulation system for which the algorithms proposed in this paper are designed consists of a robot arm in SCARA configuration, an RGB-D camera, a marker used for tracking and a manipulation software. The proposed approach is tested using a custom-made robot arm, VICRA (VIsion Controlled Robot Arm), as shown in Figure 6. VICRA has one translational and three rotational joints. The first three joints, which position the tool, are driven by stepper motors, while the fourth joint, which defines the tool orientation, is driven by a DC servo motor. The first translational joint enables vertical reach of approximately 0.6 m and the two rotational joints enable horizontal reach of approximately 0.6 m.

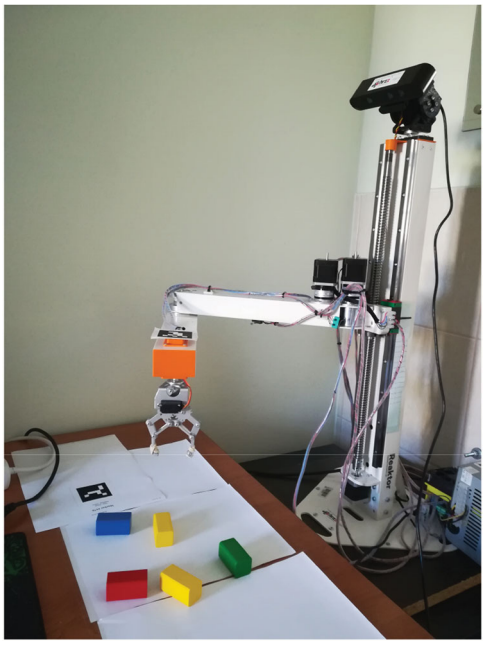

Figure 6. VICRA – robot arm in SCARA configuration.

The weight of the robot is approximately 12 kg, which makes it suitable for mounting on a mobile platform. The robot is controlled by an Arduino-based micro-controller, which communicates with a PC via USB.

An RGB-D camera, mounted on a pan-tilt head positioned at the top of the robot, observes the robot's workspace. The camera used for visual feedback is an off-the-shelf RGB-D camera, Orbec Astra S [34], optimized for short-range use cases, from 0.35 to 2.5 m which makes it suitable for smaller robots, where the camera is relatively close to objects of interest.

A gripper is mounted on the end effector and is replaced by a laser when needed. Tool positioning is achieved by tracking a marker placed on the robot's end effector, with its centre lying on the joint 4 axis, as shown in Figure 7. Marker detection and pose estimation are implemented using ArUco library for augmented reality [14,35] based on Open CV [36].

The entire setup costs below €3500. A commercial price of such robot is supposed to be even lower, since the discussed robot arm is a prototype and the development cost is included in its price.

5.2. Visual servoing experiments

The developed algorithms were experimentally tested in order to determine the accuracy of visual servoing achieved by the proposed calibration method. For this purpose, the gripper is substituted by a laser pointer. In addition to the marker placed at the end effector, another marker is used to represent the object of interest whose centre represented the target position. The

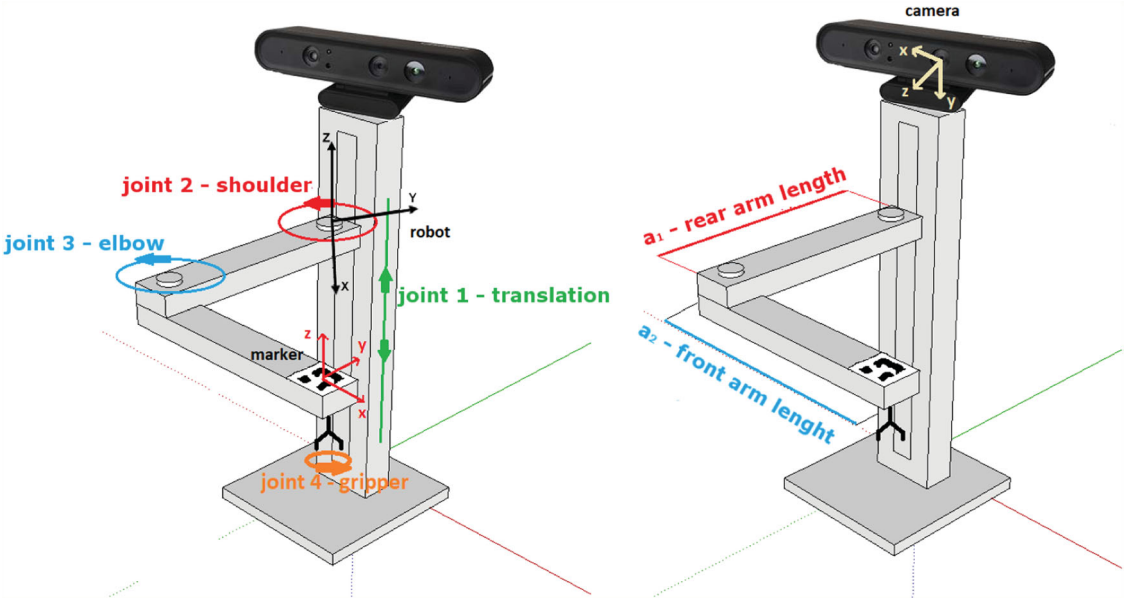

Figure 7. Vision-guided SCARA robot system.

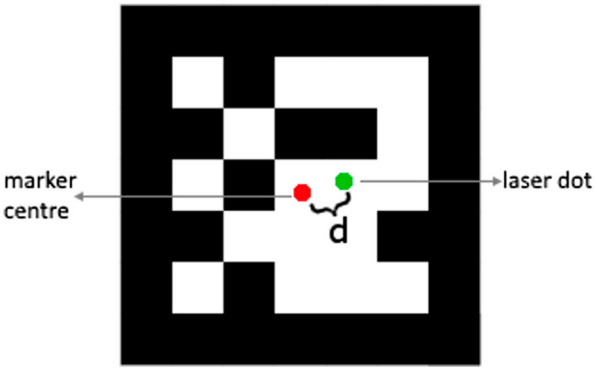

Figure 8. Distance d between the centre of the marker and the laser point.

robot arm was supposed to position the laser pointer close to this target position. After the positioning is completed, the distance d between the centre of the marker and the laser point was measured manually. An example is shown in Figure 8.

Since backlash in elbow and shoulder joints was noticeable, compensation is included each time a joint changes movement direction. Also, at the beginning of the experiment, an initial movement in positive direction for both joints must be performed. This ensures that the initial motor direction is known in order to correctly compensate for the backlash. The experiment was performed five times. Each time the camera was tilted, to guarantee a change in the relative position between the camera and the robot RF, and the two-step hand–eye calibration was performed. This process was followed by putting the marker, which represented the object of the interest, in 15 different positions in the robot's working area. Visual servoing was performed and the results are given in Table 1. This way, we tested not only accuracy of the proposed methods but their repeatability also.

Table 1. Visual servoing experimental results.

	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	Average
Average distance, $\bar{d}$ (mm)	3.47	3.00	4.00	3.87	3.93	3.65

In Figure 9, a normalized cumulative histogram is shown, where x-axis represents the distance error in millimetres, while the y-axis represents the percentage of the experiments, for which the error was below the corresponding value on the x-axis. As it can be seen, in 87.67% of the performed experiments the tool was positioned at a distance under 5 mm from the marker centre. The greatest impact on the positioning accuracy has the uncertainty in the estimation of the z-axis of the robot RF. The positioning error is proportional to the height difference between the marker placed on the robot's end effector and the marker representing the object of interest. In these evaluation experiments, this height difference was approximately 200 mm, which means that the error in z-axis estimation of 1.43° results in a positioning error of 5 mm.

5.3. Grasping experiments

In order to evaluate the applicability of the proposed grasp planning approach, a series of object grasping experiments were performed. Twelve sets of experiments were performed. Each set consisted of hand–eye calibration followed by five grasping operations. In each grasping operation, an object placed on the horizontal plane in the robot's working region, as shown in Figure 6, was detected in the scene, and its centroid, representing the target position for visual servoing, as well as tool rotation required for successful grasp are computed as described in Section 4. The rotation angle of the considered gripper mounted on the robot arm is

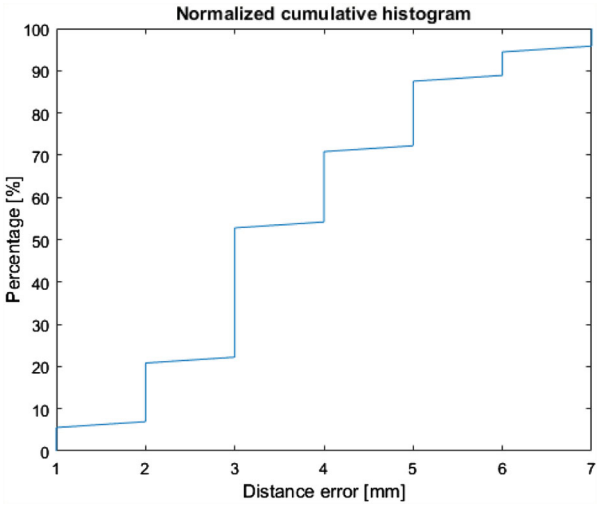

Figure 9. Normalized cumulative histogram of tool positioning error.

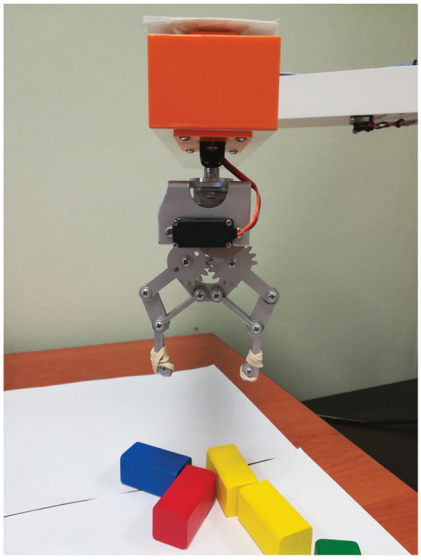

Figure 10. The initial position of the gripper.

defined on the interval $q_4 \in [0, \pi]$. Figure 10 represents the initial position of the gripper, when $q_4 = 0$. Visual servoing navigated the robot arm above the object of interest and grasping was performed. The object was finally moved to a target destination, which in this experiment was represented by a marker placed in the scene.

An experiment was considered successful if an object was properly detected, the robot arm was positioned above the object, the gripper was correctly rotated and the object was grasped, lifted and moved to the target position. The results are shown in Table 2. Out of 60 grasping experiments, 4 were unsuccessful due to the error in object recognition. Since object recognition is not the topic of this paper, failures in object recognition weren't included in the reported statistics. Three grasping operations were unsuccessful due to the insufficient visual servoing precision. In these three experiments, the object wasn't correctly grasped and it slipped off the gripper. The rest of the experiments were successful.

Table 2. Grasping experimental results.

	Successful	Unsuccessful
Number of experiments	53	3
Per cent	94.64	5.36

6. Conclusion and future work

In this paper, a vision-guided robot manipulation system is described, which uses only visual feedback for positioning of the tool and grasping of simple objects, therefore making it suitable for low-cost systems without encoders. The described system is based on visual servoing, which uses a novel fast hand–eye calibration method. A short execution time of the proposed method is of great importance when frequent recalibration is needed. The reported experimental tests prove that the obtained positioning accuracy is suitable for object manipulation tasks where accuracy of 7 mm is sufficient. Grasping was successful in 95% of experiments.

The positioning accuracy considerably depends on the accuracy of the supporting plane estimation, where the positioning error increases linearly with the distance between the marker and the tool. Hence, the positioning accuracy could be improved by a robot design, which would reduce this distance. Furthermore, it was noticed that RGB and depth registration in images captured by the considered camera are not accurate. Since the methods considered in this paper use both RGB and depth information, and therefore depend on well-aligned images, incorrect registration represents a source of the inaccuracy in positioning. One solution to this problem is to use a different RGB-D sensor with more accurate registration between the RGB and depth images, or a calibration algorithm which provides optimal registration parameters between the RGB and depth camera. In the future, a system could perform automatic recalibration each time when the camera changes its angle and point of view. Also, an extended Kalman filter for pose estimation may be included. Since few failures in object manipulation were recorded, a vision system can be used to recover the robot from failure. Failure detection and recovery strategies are also a possible topic of our future research.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work has been fully supported by the Hrvatska Zaklada za Znanost (Croatian Science Foundation) under the project number IP-2014-09-3155.

ORCID

R. Cupec http://orcid.org/0000-0003-4451-7952

References

[1] International Federation of Robotics [Internet]. [cited 2018 Feb 13]. Available from: https://ifr.org/
[2] Kim HS, Min JK, Song JB. Multi-DOF counterbalance mechanism for low-cost, safe and easy-usable robot arm. In: 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI); 2014. p. 185–188.
[3] Larouche BP, Zhu ZH. *Autonomous robotic capture of non-cooperative target using visual servoing and motion predictive control*. Auton Robots. 2014;37(2):157–167.
[4] Durović P, Grbić R, Cupec R, et al. Low cost robot arm with visual guided positioning. In: MIPRO Proceedings; Rijeka; 2017. p. 1332–1337.
[5] Craig JJ. Introduction to robotics: mechanics and control. 3rd ed. Upper Saddle River (NJ) Pearson Prentice Hall; 2005.
[6] Van Delden S, Hardy F. Robotic eye-in-hand calibration in an uncalibrated environment. J Syst. 2009;6(6):67–72.
[7] Lippiello V, Siciliano B, Villani L. Eye-in-hand/eye-to-hand multi-camera visual servoing. In: 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference, CDC-ECC'05; Seville, Spain. 2005. p. 5354–5359.
[8] Qiao Y, Liu QS, Liu GP. A new method of self-calibration of hand-eye systems based on active vision. In: The International Federation of Automatic Control; Aug 24–29; Cape Town, South Africa; 2014. p. 9347–9352.
[9] Xu H, Wang Y, Chen W, et al. A self-calibration approach to hand-eye relation using a single point. In: International Conference on Information and Automation, ICIA 2008; Zhangjiajie, China. 2008. p. 413–418.
[10] Miseikis J, Glette K, Elle OJ, et al. Automatic calibration of a robot manipulator and multi 3d camera system; 2016. arXiv preprint arXiv:1601.01566.
[11] Heller J, Henrion D, Pajdla T. Hand-eye and robot-world calibration by global polynomial optimization. In: 2014 IEEE International Conference on Robotics and Automation (ICRA); Hong Kong, China. 2014. p. 3157–3164.
[12] Strobl KH, Hirzinger G. Optimal hand-eye calibration. In: IEEE/RSJ International Conference on Intelligent Robots and Systems; Stockholm, Sweden. 2006. p. 4647–4653.
[13] Boby RA, Saha SK. Single image based camera calibration and pose estimation of the end-effector of a robot. In: IEEE International Conference on Robotics and Automation (ICRA); 2016. p. 2435–2440.
[14] Freeman RM, Julier SJ, Steed AJ. A method for predicting marker tracking error. In: 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2007; Nara, Japan. 2007. p. 157–160.
[15] Speth J, Morales A, Sanz PJ. Vision-based grasp planning of 3D objects by extending 2D contour based algorithms. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2008; Nice, France. 2008. p. 2240–2245.
[16] Lin C-C, Gonzalez P, Cheng M-Y, et al. Vision based object grasping of industrial manipulator. In: 2016 International Conference on Advanced Robotics and Intelligent Systems (ARIS); Taipei, Taiwan. 2016. p. 1–5.
[17] Xue Z, Dillmann R. Efficient grasp planning with reachability analysis. In: Liu H, Ding H, Xiong Z, Zhu X, editor. Intelligent robotics and applications. ICIRA 2010; Berlin: Springer; 2010. (Lecture notes in computer science; 6424).
[18] Marton Z-C, Pangercic D, Blodow N, et al. General 3D modelling of novel objects from a single view. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); Taipei, Taiwan. 2010. p. 3700–3705.
[19] Aldoma A, Tombari F, Di Stefano L, et al. A global hypothesis verification method for 3d object recognition. In: European Conference on Computer Vision (ECCV); Florence, Italy. 2012.
[20] Papazov C, Burschka D. An efficient RANSAC for 3D object recognition in noisy and occluded scenes. In: Asian Conference on Computer Vision (ACCV), Part I; 2010. p. 135–148.
[21] Hinterstoisser S, Cagniart C, Ilic S, et al. *Gradient response maps for real-time detection of texture-less objects*. IEEE Trans Pattern Anal Mach Intell. 2012;34(5):876–888.
[22] Mueller CA, Pathak K, Birk A. Object shape categorization in RGBD images using hierarchical graph constellation models based on unsupervisedly learned shape parts described by a set of shape specificity levels. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); Chicago (IL), USA; 2014. p. 3053–3060.
[23] Detry R, Ek CH, Madry M, et al. Learning a dictionary of prototypical grasp-predicting parts from grasping experience. In: 2014 IEEE International Conference on Robotics and Automation (ICRA); Hong Kong, China. 2014. p. 3157–3164.
[24] Twardon L, Ritter H. Interaction skills for a coat-check robot: identifying and handling the boundary components of clothes. In: IEEE International Conference on Robotics and Automation (ICRA); Seattle, Washington, USA. 2015. p. 3682–3688.
[25] Kragic D, Miller AT, Allen PK. Real-time tracking meets online grasp planning. In: IEEE International Conference on Robotics and Automation, Proceedings 2001 ICRA, Vol. 3; Seoul, Korea. 2001. p. 2460–2465.
[26] Lopez-Damian E, Sidobre D, Alami R. Grasp planning for non-convex objects. In: International Symposium on Robotics, Vol. 36; Tokyo, Japan. 2005. p. 167.
[27] Simeon T, Laumond J-P, Lamiraux F. Move3d: a generic platform for motion planning. In: Proceedings of the 4th International Symposium on Assembly and Task Planning (ISATP'2001); Fukoka, Japan. 2001.
[28] Borst C, Fischer M, Hirzinger G. Calculating hand configurations for precision and pinch grasps. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002. Vol. 2; Lausanne, Switzerland. 2002. p. 1553–1559.
[29] Miller AT, Allen PK. Graspit! a versatile simulator for robotic grasping. IEEE Robot Autom Mag. 2004;11(4):110–122.
[30] Fischler MA, Bolles RC. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Graph Image Process. 1981;24(6):381–395.
[31] Serdar K, Bingul Z. Robot kinematics: forward and inverse kinematics. In: Cubero S, editor. Industrial robotics: theory, modelling and control; 2006. ISBN: 3-86611-285-8, InTech.

[32] Cupec R, Filko D, Nyarko EK. Segmentation of depth images into objects based on local and global convexity. In: European Conference on Mobile Robotics (ECMR); Paris; 2017.
[33] Cupec R, Nyarko EK, Filko D, et al. Global localization based on 3d planar surface segments. In: Proceedings of the Croatian Computer Vision Workshop, (CCVW) 2013; Zagreb; 2013. p. 31–36.
[34] Orbbec Astra [Internet]. [cited 2017 Nov 13]. Available from: https://orbbec3d.com/.
[35] ArUco: a minimal library for augmented reality applications based on OpenCV [Internet]. [cited 2017 November 13]. Available from: https://www.uco.es/investiga/grupos/ava/node/26.
[36] Kaehler A, Bradski G. Learning OpenCV 3. Sebastopol (CA): O'Reilly Media; 2016.

Appendices

Appendix 1. Marker positioning using an RGB-D camera

The following steps explain obtaining the 3D position of the marker.

Step 1: The RGB-D camera captures an image.

Step 2: The centre of the marker is identified in the RGB image and its x-, y- and z-coordinates are computed using appropriate computer vision software.

Step 3: The coordinates of the marker centre in the depth image are determined using its coordinates in the RGB image.

Step 4: In order to reduce the impact of the measurement noise, the average depth of a square neighbourhood around the marker centre is computed. A new z-coordinate of the marker centre w.r.t. the camera is obtained.

Step 5. Scaling factor s is computed by Equation (2).

Step 6. The final 3D position of the marker is computed by multiplying each coordinate, x,y and z, computed in Step 2 with scaling factor s.

Appendix 2. Hand–eye calibration method

The following steps represent the proposed simple, two-step hand–eye calibration algorithm.

Step 1: Get a depth image from the camera, where each image point is assigned its 3D coordinates in the camera's RF.

Step 2: Compute the parameters of the dominant plane in the scene using a standard RANSAC procedure [30]. Define the z-axis of the robot RF w.r.t. the camera RF as the unit vector perpendicular to this plane.

Step 3. In the captured image, search for the marker on the end effector. The centre of the marker position represents point ${}^{C}p_{M}(0)$.

Step 4. Perform a rotation with the shoulder joint for a known angle difference Δq_2, obtained by Equation (6), in a positive direction. The camera captures an image. The new marker centre position represents point ${}^{C}p_{M}(1)$.

Step 5. Compute the position of the origin of the robot RF ${}^{C}t_{R}$ w.r.t. the camera RF by solving Equation (9).