SCALAR - Simultaneous Calibration of 2D Laser Range Finder Extrinsic Parameters And Robot Kinematics Using Three Planar Constraints

Teguh Santoso Lembono, Francisco Suárez-Ruiz, and Quang-Cuong Pham

***Abstract*—**

I. Introduction

Robots have been used in many industrial applications, such as pick and place, spray-painting, and spot-welding. In those applications, the robots either do not need very high accuracy (such as in material handling for large objects), or the robots are programmed by teaching, where the important parameter is the repeatability of the robot instead of the accuracy. Repeatability refers to the capability of the robot to return to the same location as previously taught precisely, whereas accuracy refers to the capability of the robot to reach a location (computed based on the robot kinematics model) in the robot workspace. If the robots are taught by teaching pendants to reach certain locations, the accuracy of the robot becomes irrelevant to the task.

However, there are a lot of interests to use robot on many other applications where the accuracy of the robot becomes very crucial, in which the robot has to adapt to each task automatically with a great precision. Consider, for example, a robot drilling task. In robot drilling task, the robot is supposed to drill several holes at precisely-defined locations on a workpiece, which can change for each task. The position of the holes can be taught manually for each workpiece, but that will take a lot of time and effort. If we want to program the robot automatically for such a task, the robot has to be able to do a few things accurately: the robot has to scan the workpiece, determine the location for the holes, and finally move to that location precisely. The accuracy of such a robotic system depends on several factors: The accuracy of the robotic arm, the accuracy of the measurement system, and the accuracy of the end effector tool transformation.

The accuracy of the robot arm is determined by how closely the kinematic parameters of the robot model resembles the actual kinematic parameter of the physical robot. This is affected by the manufacturing process, the assembly process, and the wear and tear during the operation of the robot. To achieve a better accuracy, a robot calibration is normally conducted, either by using an external measurement system (such as motion capture system, or spinarm, or CMM), or by constraining the motion of the end effector.

The accuracy of the measurement system can be divided into two parts: the accuracy of the measurement device itself, and the accuracy of the homogeneous transformation between the measurement frame to the robot frame (often called the extrinsic parameters of the measurement system). A camera system, for example, is generallly less precise as compared to a laser system, although a camera can give more information. To obtain the accurate extrinsic parameters of a measurement device, the information from the CAD model is not sufficient as there will be error during the manufacturing and the assembly, and very often the measurement coordinate is located inside the measurement device, causing it to be difficult to be measured accurately. When the measurement device is attached to the robot, a process known as hand-eye calibration is normally used to find the extrinsic parameters.

Lastly, the accuracy of the end effector transformation refers to the homogeneous transformation from the robot flange coordinate frame to a pre-determined location on the end effector (such as the tip of the drilling bit). Calibration of the end effector is generally known as TCP calibration.

In this paper we concern ourselves with solving the first two issues. We propose to use a near-range 2D Laser Range Finder (LRF), attached on a robot arm, to calibrate the kinematics parameters of the robot and at the same time also calibrate the extrinsic parameters of the LRF. The LRF is chosen because it can give very accurate measurement data, both for the calibration and for the subsequent task (such as drilling). The proposed method is much easier to be implemented as compared to the previous methods, and it does not require expensive measurement device.

The overall calibration procedure is as follows:

1) The scanner is attached on the robot, and three (approximately) perpendicular planes are set around the robot. The location (position and orientation) of the planes only needs to be known approximately.
2) The robot is moved to several positions, such that the LRF's 2D ray intersects each of the plane (one at a time). The data from the LRF as well as the joint angles information are collected.
3) An optimization algorithm (Levenberg-Marqueadt) is used to find the optimal robot kinematics parameters, together with the LRF extrinsic parameters, by satifsfing the following constraints: the projected LRF data should fall on the three planes.

The remainder of the paper is as follows. In Section II, we discuss the existing approaches to the calibration problem, both for the robot kinematics and the LRF extrinsic parameters, and how our proposed method differs from the others. In Section III, the proposed method is explained in

The authors are with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore.

detail. A simulation study is presented in Section IV to verify the method, and finally we conclude the paper with a few remarks in Section V.

II. Related works

A. Calibration of robot kinematic parameters

Robot kinematics calibration have been researched for quite a long time; some of the earliest work began in 1980s. Generally, the calibration process can be divided into unconstrained and constrained calibration. In unconstrained calibration, the robot moves its end-effectors to several poses freely, while an external measurement system measures the pose. The measured pose is then compared to the one computed from the kinematics model, and the model can be updated to minimize the difference between the predicted pose and the measured pose.

For example, Ginani and Mota [1] calibrate an ABB IRB 2000 industrial robot using a ROMER measurement arm, which improves the mean/maximum position errors from 1.25mm/2.20mm to 0.30mm/1.40mm. Ye et al. [2] used a Faro laser tracker to calibrate an ABB IRB 2400/L, and improve the mean/maximum position errors from 0.963mm/1.764mm to 0.470 mm/0.640mm. In [3], Nubiola and Bonev used a Faro laser tracker ION to calibrate an ABB IRB 1600-6/1.45 robot. The mean/maximum position errors are reduced from 0.968mm/2.158mm to 0.364 mm/0.696 mm, respectively.

The issue with such calibration process is the difficulty in setting up the calibration setup and the expensive cost of the external measurement system. For example, the cost of a laser tracker is more than $\$100,000$[3]. Therefore, many researchers try to find calibration methods which only rely on the internal sensors of the robot, such as by constraining the motion of the end-effector to provide the calibration equations.

In [4], Ikits and Hollerbach propose a kinematic calibration method using a planar constraint. The robot end effector (a touch probe) is moved to touch random points on a plane. When the touch probe is in contact with the plane, the joint angles are recorded. The kinematics parameters of the robot model are then updated to satisfy the planar constraint. While the approach is promising, they also report observability problems for some of the calibration parameters.

In [5], Zhuang et al. investigate robot calibration with planar constraints, in particular the observability conditions of the robot kinematic model parameters. They prove that a single-plane constraint is insufficient for calibrating a robot,and a minimum of three-plane constraint is necessary. Using a three-plane constraint, the constraint system is proved to be equivalent to an unconstrained point-measurement system under three conditions: a) All three planes are mutually non-parallel, b) the identification Jacobian of the unconstrained system is nonsingular, and c) the measured points from each individual plane do not lie on a line on that plane. They verified the theory by doing a simulation on a PUMA560 robot.

In [6], Joubair and Bonev calibrated both the kinematic and non-kinematic (stiffness) parameters of a FANUC LR Mate 200iC industrial robot by using planar constraints, in the form of a high precision 9-in. granite cube. The robot is equipped with an MP250 Renishaw touch probe, which is then moved to touch four planes of the granite cube. The granite cube's face is flat to within 0.002mm. They improved the maximum plane error from 3.740mm to 0.083mm.

B. Calibration of extrinsic 2D LRF parameters

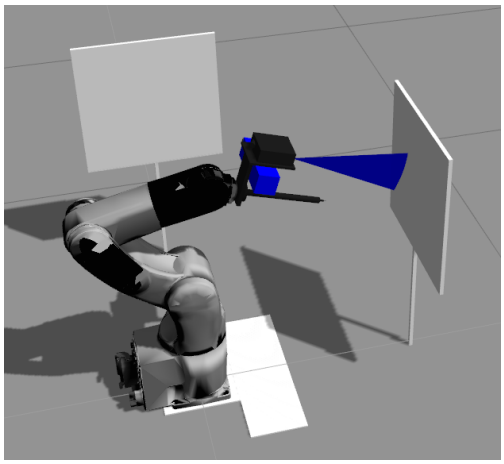
Extrinsic calibration of an LRF consists of finding the correct homogeneous transformation from the laser coordinate frame to the robot coordinate frame. While it is similar to the hand-eye calibration of a robot-camera system, the data obtained from an LRF is less informative than a camera, which makes it slightly more difficult. A camera on a robot is normally calibrated by using a checkerboard pattern, the pose of which (position and orientation) with respect to the camera can be obtained by the camera using just a single image. In contrast, we cannot obtain the pose of an object easily using just a single data from an LRF.

Most of the works on extrinsic calibration of an LRF involves a camera, since both sensors are often used together in many applications. The works are mostly based on Zhang and Pless' work[7]. They propose a method to calibrate both a camera and an LRF using a planar checkerboard pattern. First, the camera is calibrated by using a checkerboard pattern, using a standard hand-eye calibration procedure. The calibrated camera is then used to calculate the pose of the pattern. Next, the robot is moved to several poses with the LRF pointing to the pattern. By using the geometric constraints that all the data points from the LRF should fall on the pattern plane, the extrinsic parameters of the LRF can be obtained. Finally, the same constraints is used to optimize both the intrinsic and extrinsic parameter of the camera and the extrinsic parameter of the LRF. The nonlinear optimization problem is solved by using the Levenberg-Marquardt optimization algorithm.

Unnikrishnan and Hebert [8] used the same setup as [7], although they do not optimize the camera parameter simultaneously due to the nonlinearity of the resulting cost function. Li et al. [9] used a specially designed checkerboard to calibrate the extrinsic parameters between a camera and an LRF, and claim that the result is better than [7]. Vasconcelos et al. [10] developed a minimal closed-form solution for the extrinsic calibration of a camera and an LRF, based on the work in [7].

Our proposed algorithm can be seen as a combination of the algorithm for extrinsic calibration of LRF [7] and the algorithm for calibration of the robot kinematics parameter using three planar constraints [6]. The proposed method simultaneously optimize both the LRF extrinsic parameters and the robot kinematics parameters to satisfy the planar constraints. It has the following advantages as compared to the other calibration approaches:

1) The method does not need an additional camera to calibrate the LRF, unlike [7]



(a)

Fig. 1. Robot setup for the calibration

2) The method does not need another expensive external measurement system. The measurement is done using the LRF which will also be used in the robot task, so it does not incur additional cost. Moveover, an LRF can achieve very high accuracy at much lower cost as compared to measurement system such as Vicon or Faro Laser Tracker.
3) The method does not need a precisely manufactured calibration object such as the granite cube in [6], which requires the planes' location and orientation to be known accurately. The proposed method only requires three flat surfaces which are oriented somewhat perpendicularly, and the location and orientation need to be known only approximately.
4) The calibration setup can be done very easily, since the planes do not need to assume precisely known locations.
5) The calibration poses can be distributed throughout the whole workspace, instead of being confined only in a local region like in [6].

III. Method

The calibration setup is depicted in Fig. 1, where three perpendicular planes ($k = 1, 2, 3$) are installed around the robot. An LRF is attached on the robot flange. For each plane, the robot is moved to N poses such that the LRF is directed to the respective plane. One data set from the LRF consists of hundreds of data points, so M data points are selected randomly from the LRF data for each pose, together with the joint angles of the robot.

This section provides the detail on how to calibrate both the extrinsic parameters of the LRF and the robot kinematics parameters. First, the initial estimate of the LRF extrinsic parameter is obtained by using the linear least-square method based on the data from one of the planes. After that, the LRF extrinsic parameters and the robot kinematics parameters are optimized simultaneously to satisfy the three planar constraints, by using the Levenberg Marquardt nonlinear optimization method. At the end of this section, we explain how we can use Singular Value Decomposition (SVD) to analyse which calibration parameters are identifiable, and the steps to handle the unidentifiable parameters are then presented.

A. Obtaining the Initial Estimate of the LRF Extrinsic Parameters

To obtain an initial estimate of the LRF extrinsic parameters, only the data from one plane is necessary. Arbitrarily, the bottom plane $plane1$ is chosen. The extrinsic parameters of the LRF ${}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}}$, i.e. the homogeneous transformation from the robot flange coordinate frame to the LRF coordinate frame, is estimated by the following calculation.

Let p_i be one of the data points from the LRF which lies on the $plane1$, n_1 be the normal unit vector of $plane1$, and d_1 be the perpendicular distance from the origin of the robot coordinate system to $plane1$. The subscript/superscript B, E, and L denote the coordinate frame of the robot base, robot flange, and the LRF, while the subscript i and j refer to the data point index and the robot pose index respectively. Since p_1 is located on the $plane1$, it has to satisfy the following equation:

$$({}^{\boldsymbol{B}}\boldsymbol{n_1})^T({}^{\boldsymbol{B}}\boldsymbol{p_i}) - {}^{B}d_1 = 0 \tag{1}$$

${}^{\boldsymbol{B}}\boldsymbol{p_i}$ depends on the robot pose ${}^{\boldsymbol{B}}\boldsymbol{T}_{\boldsymbol{E,j}}$ at pose index j and the LRF extrinsic parameter ${}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}}$, so equation 1 can be expanded:

$$({}^{\boldsymbol{B}}\boldsymbol{n_1})^T({}^{\boldsymbol{B}}\boldsymbol{T}_{\boldsymbol{E,j}})({}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}})({}^{\boldsymbol{L}}\boldsymbol{p_i}) - {}^{B}d_1 = 0 \tag{2}$$

${}^{\boldsymbol{B}}\boldsymbol{n_1}$ and ${}^{B}d_1$ are known approximately ($[0\ 0\ 1\ 0]$ and 0.0), ${}^{\boldsymbol{B}}\boldsymbol{T}_{\boldsymbol{E,j}}$ can be computed from the robot's joint angles at pose j, and ${}^{\boldsymbol{L}}\boldsymbol{p_i}$ is obtained from the laser. Let $\boldsymbol{n_j^{'}} = ({}^{\boldsymbol{B}}\boldsymbol{n_1})^T({}^{\boldsymbol{B}}\boldsymbol{T}_{\boldsymbol{E,j}}) = \begin{bmatrix} n_{j,1}^{'} & n_{j,2}^{'} & n_{j,3}^{'} & n_{j,4}^{'} \end{bmatrix}$, then

$$(\boldsymbol{n_j^{'}})^T({}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}})({}^{\boldsymbol{L}}\boldsymbol{p_i}) - {}^{B}d_1 = 0 \tag{3}$$

The only unknown in equation 3 is ${}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}}$ which has 12 elements r_{uv}, where u and v denote the column and the row index of the matrix. Note that the fourth row only consists of 0 and 1. Without any loss of generality, let's assume that the data points from the LRF lies on the XZ planes in the laser frame L, so ${}^{\boldsymbol{L}}\boldsymbol{p_i} = \begin{bmatrix} {}^{L}p_{i,x} & 0 & {}^{L}p_{i,z} & 1 \end{bmatrix}$. If we expand the equation above and rearrange such that the components of ${}^{\boldsymbol{E}}\boldsymbol{T}_{\boldsymbol{L}}$ are gathered as a vector $\boldsymbol{\Phi_L}$, we have

$$(\boldsymbol{x_{ji}})^T(\boldsymbol{\Phi_L}) = {}^{B}d_1 - n_{j,4}^{'} \tag{4}$$

where

$$\boldsymbol{x_{ji}} = \begin{bmatrix} {}^{L}p_{i,x}n_{j,1}^{'} & {}^{L}p_{i,x}n_{j,2}^{'} & {}^{L}p_{i,x}n_{j,3}^{'} & {}^{L}p_{i,z}n_{j,1}^{'} \\ {}^{L}p_{i,z}n_{j,2}^{'} & {}^{L}p_{i,z}n_{j,3}^{'} & n_{j,1}^{'} & n_{j,2}^{'} & n_{j,3}^{'} \end{bmatrix} \tag{5}$$

, and

$$\boldsymbol{\Phi_L} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & r_{13} & r_{23} & r_{33} & r_{14} & r_{24} & r_{34} \end{bmatrix} \tag{6}$$

For each data point i, we obtain such equation as in equation 4. For M data points per pose and a total of N robot poses, we have MN such equations. The equations can be stacked together to form the following matrix equation:

$$\boldsymbol{X\Phi_L = D} \tag{7}$$

where

$$X = \begin{bmatrix} x_{11} \\ x_{12} \\ \cdots \\ x_{1M} \\ x_{21} \\ \cdots \\ x_{2M} \\ \cdots \\ x_{NM} \end{bmatrix}, \; D = \begin{bmatrix} {}^{B}d_1 - n'_{1,4} \\ {}^{B}d_1 - n'_{1,4} \\ \cdots \\ {}^{B}d_1 - n'_{1,4} \\ {}^{B}d_1 - n'_{2,4} \\ \cdots \\ {}^{B}d_1 - n'_{2,4} \\ \cdots \\ {}^{B}d_1 - n'_{N,4} \end{bmatrix} \tag{8}$$

Equation 7 can be solved by linear least square procedure to obtain $\boldsymbol{\Phi_L}$. $\boldsymbol{{}^{E}T_L}$ can then be computed as follows:

1) The parameters $[r_{11}, r_{21}, r_{31}]$ and $[r_{13}, r_{23}, r_{33}]$ are supposed to be unit vectors, so they should be normalized. They consistute the first and the third column of the matrix ${}^{E}T_L$
2) The parameters $[r_{14}, r_{24}, r_{34}]$ constitutes the position component of the matrix ${}^{E}T_L$, or its fourth column
3) The parameters $[r_{12}, r_{22}, r_{32}]$ can be calculated as the cross product of $[r_{13}, r_{23}, r_{33}]$ and $[r_{11}, r_{21}, r_{31}]$

To use the extrinsic parameters $\boldsymbol{{}^{E}T_L}$ for the subsequent step, 9 parameters would be too redundant. Therefore, we choose the axis-angle representation for the rotation part of ${}^{E}T_L$ which gives us 4 parameters $[r_1, r_2, r_3, r_4]$, and 3 more parameters for the location part $[p_x, p_y, p_z]$.

B. Optimizing both the LRF Extrinsic Parameters and Robot Kinematic Parameters

In the second step, the data from all the three planes are used to optimize the parameters of the LRF, robot kinematics, as well as the plane parameters. The objective function is described as follows:

$$\mathbf{f}(\boldsymbol{\Phi}) = \sum_{k=1}^{3}\sum_{j=1}^{N}\sum_{i=1}^{M}((\boldsymbol{{}^{B}n_1})^T(\boldsymbol{{}^{B}p_i}) - \boldsymbol{{}^{B}d_1})^2 \tag{9}$$

The parameters $\boldsymbol{\Phi}$ consists of the followings:

1) Robot kinematics parameters. We choose the modified DH parameters [11] $[a_i\ , \alpha_i\ , \theta_i\ , d_i], i = 1, 2, \cdots, 6$ to represent the robot kinematics, so we have 6*4 = 24 DH parameters for a 6 DoF robot arm
2) Laser extrinsic parameters. As mentioned in the previous section, we use the axis angle representation for the rotation part $[r_1, r_2, r_3, r_4]$,and the location is represented by 3 numbers $[p_x, p_y, p_z]$.
3) Plane parameters. Each plane has 4 parameters $[n_{k,x}, n_{k,y}, n_{k,z}, d_k]$, 3 for the unit vector and 1 for the distance, so we have 4*3 = 12 parameters.

In total, we have 43 parameters to be optimized by minimizing the objective function $f\Phi$. To do that, we need to ensure that the number of data points MN exceeds the number of parameters. The optimization problem is then solved by using Levenberg-Marquardt nonlinear optimizer [12]. For the unit vector parameters ($[r_1, r_2, r_3]$ and $[n_{k,x}, n_{k,y}, n_{k,z}]$), the following constraints are added to the solver:

$$r_3 = \sqrt{1 - r_1^2 - r_2^2} \tag{10}$$

$$n_{k_z} = \sqrt{1 - n_{k_x}^2 - n_{k_y}^2} \tag{11}$$

The objective function $\boldsymbol{f(\Phi)}$ basically uses the constraints that all data points from the LRF have to be on the respective plane. [5] proved that such constraints are essentially equivalent to the calibration of a robot using end-point measurement. Further analysis on the observability of the parameters will be presented on the next section.

C. The identifiability of the calibration parameters

Depending on the chosen robot calibration poses and the choices of the parameters, some of the calibration parameters might not be observable, due to the redundancy among the parameters. This is a critical problem in calibration, as it will result in some of the parameters assuming erratic values and unstable calibration result. To prevent that, we have to first analyse which calibration parameters are identifiable and which are not.

Following the approach in [13] and [6], SVD is applied on the identification Jacobian matrix J. The matrix J can be computed as follows. Let $f_{kji}(\Phi)$ be the constraint equation on one of the data point i at the robot pose j and on the plane k:

$$f_{kji}(\Phi) = (\boldsymbol{{}^{B}n_k})^T(\boldsymbol{{}^{B}p_i}) - \boldsymbol{{}^{B}d_k} \tag{12}$$

Then J can be computed by differentiating all the data points $i = 1, \cdots, M$ for all the poses $j = 1, \cdots, N$ and for all the planes $k = 1, 2, 3$, and stack them together as a matrix:

$$\boldsymbol{J} = \begin{bmatrix} \frac{\partial f_{111}(\Phi)}{\partial \Phi} \\ \frac{\partial f_{112}(\Phi)}{\partial \Phi} \\ \cdots \\ \frac{\partial f_{3MN}(\Phi)}{\partial \Phi} \end{bmatrix} \tag{13}$$

We can then apply SVD on matrix J:

$$\boldsymbol{J = U\Sigma V^T} \tag{14}$$

Note that for this identification step, the parameters $[r_3, n_{1_z}, n_{2_z}, n_{3_z}]$ are excluded from the parameter vector Φ, since those four parameters are obtained as linear combinations of other parameters (Equation 10-11). That leaves us with 43-4 = 39 parameters in Φ, which correlates to the 39 singular numbers in Σ. The number of singular values with the value of zero in Σ is then equal to the number of unidentifiable parameters. For a given zero-value singular number σ_r, the rth column vector of matrix V is the linear combination of the parameters Φ which cannot be identified independently. In our case, out of the 39 parameters, there

are 7 redundant parameters. By analysing the matrix V, those 7 redundant parameters are due to:

1) The parameters d_6 (the translation along the z-axis of the 6th link frame on the flange) and p_z (the z coordinate of the LRF frame) are redundant. This has obvious physical meaning, because if we shift the origin of the frame 6 in its z direction, we can compensate by shifting the origin of the LRF frame in the opposite direction
2) The parameters θ_6 and $r4$ are redundant. These are the rotation of the last link and the rotation of the LRF frame, both around the same z-axis.
3) The parameters d_2 and d_3 are redundant. We can shift the frame 2 in its z direction, and compensate by shifting the frame 3 in the other direction.
4) Lastly, we have four redundant parameters as the linear combination of the DH parameter of the first link $(a_1, \alpha_1, \theta_1, d_1)$ and all the three planes' parameters. Physically, this relates to the fact that we can adjust the location of the base frame freely by changing the value of $(a_1, \alpha_1, \theta_1, d_1)$, and all the planes' parameters will adjust according to the new base location. In other words, the base coordinate is not constrained, or it is "floating"

Having identified the redundant parameters, we can eliminate them by keeping some of the parameters to be fix. In this case, we set the value of the parameters $[d_6, \theta_6, d_2, a_1, \alpha_1, \theta_1, d_1]$ to be fix, or to maintain its original model values.

IV. Simulation

The proposed method is verified through simulation of the calibration procedure. The simulation is conducted by using ROS and Gazebo, where the robot model, the LRF, and the three planes can be simulated. As shown in Fig. 1, three perpendicular planes are located around the robot, and the robot is moved such that the LRF ray intersects each of the plane. Simulated data from the LRF can be obtained, and Gaussian noise can even be added. The data is then used as input to the calibration procedure.

The robot kinematics model in the simulation is considered as having the "true" kinematics parameters, and another kinematics model is generated by introducing random gaussian errors to the true parameters within the range of $\pm$ 2mm and $\pm$ 1^o for the linear and angular parameters, respectively. Note that the errors are considered to be very big, and they are purposedly set to be large to illustrate the robustness of the calibration method.

After the calibration procedure, the robot is moved to 1000 random poses to evaluate the proposed calibration method. Let ${}^{B}\boldsymbol{T}_{\boldsymbol{L,j,true}}$ and ${}^{B}\boldsymbol{T}_{\boldsymbol{L,j,model}}$ be the true and calibrated pose of the LRF frame w.r.t. the robot base at the robot pose index j, respectively, then the error can be computed as follows.

$$\Delta T_j = ({}^{B}\boldsymbol{T}_{\boldsymbol{L,j,model}})^{-1}\,{}^{B}\boldsymbol{T}_{\boldsymbol{L,j,true}} \tag{15}$$

Let δ_{uv} be the element of ΔT_j, then the error in position, δp_j, can be computed by

$$\delta p_j = \sqrt{\delta_{14}^2 + \delta_{24}^2 + \delta_{34}^2} \tag{16}$$

. Let δR_j be the rotation part of the homogeneous transformation matrix ΔT_j. δR_j can be represented by using an axis-angle notation, $[r_{j,1}\ r_{j,2}\ r_{j,3}\ \delta\theta_j]$. $\delta\theta_j$ is the error in rotation (it can be seen as the amount of rotation necessary to turn the calibrated pose to the true pose).

By using the proposed error formulation, the initial average error (due to the random errors introduced to the model) is 21mm and 4^o.

A. The effect of measurement accuracy

The accuracy of the calibration procedure greatly depends on the accuracy of the measurement system. In this section, the calibration is conducted at different levels of LRF accuracy, as shown in Fig. 2. The accuracy is adjusted in the simulation by changing the gaussian noise. As can be seen in Fig. 2, the error decreases as the accuracy of the LRF improves. At 0.1mm accuracy (which can be achieved by an LRF very easily), the position error and the orientation error are around 0.1mm and 0.02^o, respectively.

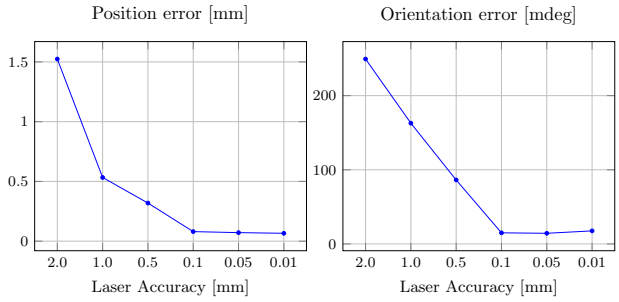


Fig. 2. Effect of the laser accuracy towards: a) Linear Accuracy b) Angular Accuracy

B. The effect of the number of calibration poses

In Fig. 3, it can be seen that as the number of poses N increases, the error decreases until around N=20, beyond which it does not change much. It can be concluded that 20 robot poses are sufficient to calibrate the robot model accurately. Note that N refers to the number of robot poses w.r.t. to one plane, so the total poses will be 3N = 60 poses for all the three planes.

In addition, we also vary M, the number of points from the LRF data per robot pose that are used for calibration 4. Note that after around 10 points, the calibration result does not improve significantly. Therefore, although an LRF can give more than 300 data points per robot pose, using more than 10-20 points do not improve the calibration by much.

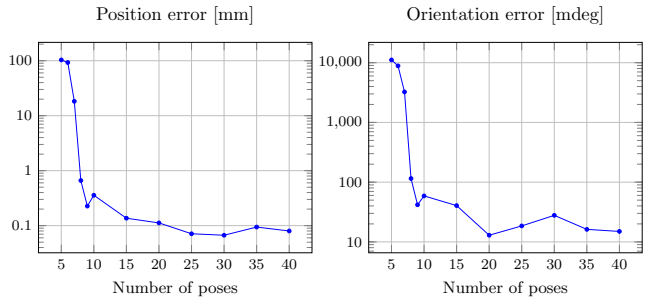


Fig. 3. Effect of the number of poses towards: a) Linear Accuracy b) Angular Accuracy

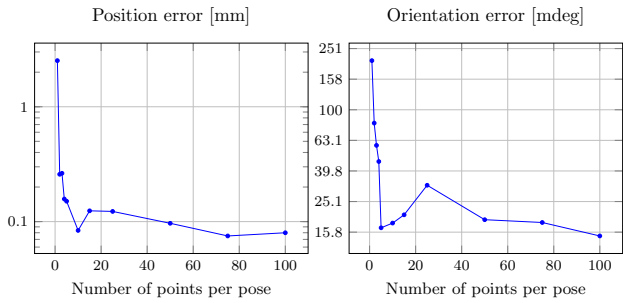


Fig. 4. Effect of the number of points towards: a) Linear Accuracy b) Angular Accuracy

C. The effect of the plane parameters' initial estimate

One of the advantage of the proposed method is that the planes parameters do not need to be precisely known. Here we vary the planes parameters estimate to demonstrate the robustness of our method. The estimated normals and position of the planes are disturbed by as much as 30^o and 65mm, as shown in Fig. 5. It is obvious that the errors in the planes' parameters estimate do not affect the calibration accuracy at all. In fact, after calibration, the planes' parameters in the calibrated model approach the true parameters very closely (within 0.1mm and 0.02^o).

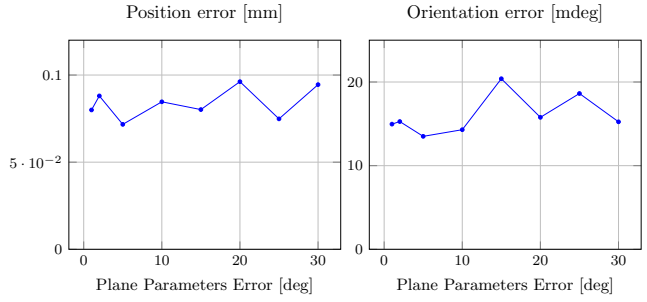


Fig. 5. Effect of the accuracy of the plane parameters initial estimate towards: a) Linear Accuracy b) Angular Accuracy

V. Conclusions

In this paper, we have proposed a method to calibrate simultaneously a 2D LRF extrinsic parameters and a 6DoF robot kinematics by using just three flat planes, arranged perpendicularly towards each other around the robot. The proposed method is much easier to implement than the previous methods in the literature, as it does not require other expensive measurement system or tedious setup. Through the simulation, we have also verified that the method can reduce the error in position and orientation from (21mm, 4^o) to (0.1mm, 0.02^o), respectively. This is very useful for many industrial robotics application which requires great accuracy. The method will be implemented on the real robot, and we will present the result in the next work.

Acknowledgment

This work was supported in part by NTUitive Gap Fund NGF-2016-01-028 and SMART Innovation Grant NG000074-ENG.

References

[1] L. S. Ginani and J. M. S. T. Motta, "Theoretical and practical aspects of robot calibration with experimental verification," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 33, no. 1, pp. 15–21, 2011.

[2] S. H. Ye, Y. Wang, Y. J. Ren, and D. K. Li, "Robot calibration using iteration and differential kinematics," *Journal of Physics: Conference Series*, vol. 48, no. 1, pp. 1–6, 2006.

[3] A. Nubiola and I. A. Bonev, "Absolute calibration of an ABB IRB 1600 robot using a laser tracker," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 1, pp. 236–245, 2013.

[4] M. Ikits and J. Hollerbach, "Kinematic calibration using a plane constraint," *Proceedings of the 1997 International Conference on Robotics and Automation*, vol. 2, no. 4, pp. 3191–3196, 1997.

[5] H. Z. H. Zhuang, S. Motaghedi, and Z. Roth, "Robot calibration with planar constraints," *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, vol. 1, pp. 1–25, 1999.

[6] A. Joubair and I. A. Bonev, "Non-kinematic calibration of a six-axis serial robot using planar constraints," *Precision Engineering*, vol. 40, pp. 325–333, 2015.

[7] Qilong Zhang and R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, no. January 2004, 2004, pp. 2301–2306.

[8] R. Unnikrishnan and M. Hebert, "Fast Extrinsic Calibration of a Laser Rangefinder to a Camera," *Robotics*, vol. 2005, no. July 2005, p. 23, 2005.

[9] G. Li, Y. Liu, L. Dong, X. Cai, and D. Zhou, "An algorithm for extrinsic parameters calibration of a camera and a laser range finder using line features," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3854–3859, 2007.

[10] F. Vasconcelos, J. P. Barreto, and U. Nunes, "A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2097–2107, 2012.

[11] S. Hayati and M. Mirmirani, "Improving the absolute positioning accuracy of robot manipulators," *Journal of Robotic Systems*, vol. 2, no. 4, pp. 397–413, 1985.

[12] M. Newville, T. Stensitzki, D. B. Allen, and A. Ingargiola, "LMFIT: Non-Linear Least-Square Minimization and Curve-Fitting for Python¶," sep 2014.

[13] J. M. Hollerbach and C. W. Wampler, "The calibration index and taxonomy for robot kinematic calibration methods," *International Journal of Robotics Research*, vol. 15, no. 6, pp. 573–591, 1996.