Kinematic Calibration of Six-Axis Robots

Kathleen English
Mechanical & Aerospace Engineering, Carleton University, Canada

Matthew John D. Hayes
Mechanical & Aerospace Engineering, Carleton University, Canada
and Institut für Automation, Montanuniversität Leoben, Austria

Markus Leitner
Institut für Automation, Montanuniversität Leoben, Austria

Christian Sallinger
Institut für Automation, Montanuniversität Leoben, Austria

Abstract: In this paper an approach to the kinematic calibration of serial robots with six revolute axes in general, and the KUKA KR-15/2 in particular, is presented. We proceed by formulating how the pose of the robot (the position and orientation of the end-effector) varies with the kinematic parameters that describe the robot geometry. The pose of a robot is specified by six independent position and orientation parameters. Because of this independence, subsets of these parameters can be used to formulate the system of calibration equations. The model presented, based on the Denavit-Hartenberg parameters, is designed for use when only position measurements are available. The measurement concept involves comparing relative linear robot motions to parallel standard ruled and flat straight edges. Images from a CCD camera mounted to the tool flange yield relative distance measurements along the length of the ruled straight edge while a stereo laser displacement sensor yields height measurements above the flat straight edge. The robot needs be calibrated only over the required workspace to perform a task, potentially enhancing task-specific accuracy. Results for simulated measurements using the three x, y, z position coordinates are presented.

Résumé: Dans cet article, une approche générale pour l'étalonnage cinématique des robots séries à six couples rotöides est présentée. Le cas particulier du KUKA KR-15/2 est aussi détaillé. Le processus consiste à formuler la variation de la pose (position et orientation de l'extrémité du manipulateur) du robot par rapport aux paramètres cinématiques qui décrivent sa géométrie. La pose d'un robot peut être décrite par six paramètres indépendants de position et d'orientation. Grâce à cette indépendance, un sous groupe de ces paramètres peut être utilisé pour formuler un système d'équations de calibration. Le modèle présenté, basé sur les paramètres de Denavit-Hartenberg, est conçu pour être utilisé seulement quand les mesures de position sont disponibles. La mesure du déplacement consiste à comparer un mouvement linéaire du robot à une règle droite graduée. Des images de la règle provenants d'une camŕa numŕique (camŕa CCD) montée sur le manipulateur donnent des mesures de distance relatives au déplacement du robot. Un capteur laser de déplacement stéréo permet de mesurer la hauteur au-dessus de la règle graduée. Des résultats provenant de simulations où les trois coordonnées de position x, y, z ont été utilisés sont présentés.

Introduction: Calibration is essential for robot manufacturing systems largely because of the robot: they normally have good repeatability, but notoriously poor accuracy. The accuracy, never stated by manufacturers, is typically an order of magnitude worse than the repeatability, which is the usual positioning performance indicator specified by manufacturers. The calibration systems reviewed require *absolute measurements* of the position of a reference point on the robot and/or orientation of the end-effector.

In this paper we present a novel measurement and calibration concept that relies instead on *relative measurements* which requires no measurement by external devices to assess the robot pose (position and orientation). This is, in a sense, more natural as robot motions are always computed relative to the current pose. The calibration model requires changes in pose as inputs. We propose to use the relative motions of the robot together with corresponding images of the ruled straight edge and changes in distance from the flat straight edge to supply the differences directly. Thus, the robot-mounted camera will capture an image of a thermal-dimensionally stable ruled standard, while a laser distance sensor will determine the height above a parallel flat standard for a set of poses that have the robot move linearly in the direction of the straight edges, a first prototype is shown in Figure 1. In principle, the difference between the position of rulings in adjacent images, together with the difference in height above the flat standard, are the only required measurements. The robot needs be calibrated only over the required workspace to perform a task, potentially enhancing task-specific accuracy. Between 50 and 100 measurements are required, depending on the desired complexity of the calibration model.

Before proceeding, we leisurely examine the history of robot calibration to put our contribution in context. Robots have been used in the manufacturing industry for many years. The first industrial robot, the famous Unimate, was sold to General Motors and installed in New Jersey in 1961. Obeying step-by-step commands stored on a magnetic drum, the 3,000-kg arm sequenced and stacked hot pieces of die-cast metal. Precision and accuracy were not unduely large concerns (Mooring et al., 1991). These robots were relatively easy to program, as all positions were taught.

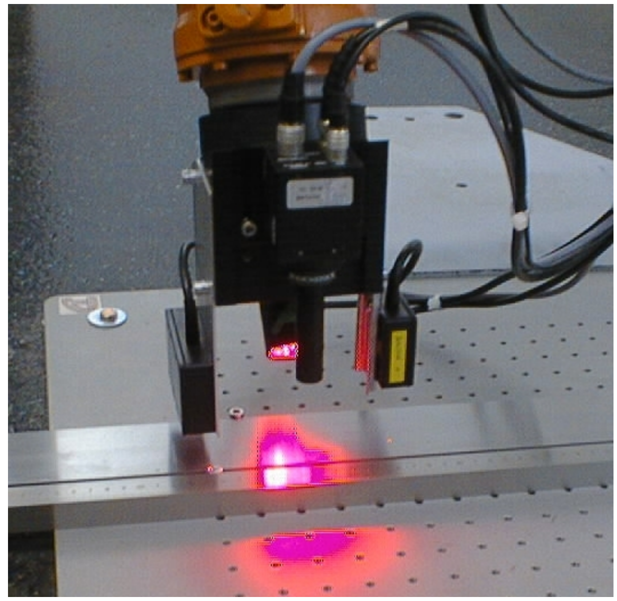

Figure 1. KUKA KR-15/2 Calibration Measurement Head.

In these early applications, repeatability was an important issue, which is the ability of that robot to position the end-effector in the same place, with the same orientation, repeatedly. Increasingly, there was demand for more intelligent robots to complete more complicated tasks than those involving repeatable pick-and-place motions: such as the need for a robot to accurately follow a continuous path in seam welding operations; or the insertion of large numbers of electronic components in printed circuit boards.

As manufacturing technology improved, so demands on robot accuracy increased. Substantial errors were noticed between where the robot was commanded to go and where the robot controller sent it, namely, the difference between robot repeatability and accuracy had become apparent. Here, accuracy refers to the ability of a robot to position itself into any arbitrary commanded pose, rather than return to the same pose many times. It has also become apparent that small deviations from the nominal robot geometric parameters could produce large errors at the end-effector. Hence, a need for the improvement in the accuracy of a robot has become paramount.

Experience has shown that the greatest portion of the robot inaccuracy is contributed by kinematic, or geometric errors. Corrections in the accuracy of these errors can produce improvements to the same order of magnitude as the repeatability (Mooring et al., 1991). Improvements in manufacturing tools and techniques yielding tighter tolerances and stiffer mechanical structures have been shown to reduce the geometric errors in robots, but come at a great expense in time and money. Since nothing can be manufactured to exactly match the specified dimensions, there will still be errors. Machine tool manufacturers, therefore, saw an economic advantage to developing calibration systems. One early calibration method was known as machine calibration, which uses internal compensation software on an individual machine basis to compensate for absolute accuracy deficiencies, resulting in a high-level of repeatability. Later research led to high precision measurement systems that include coordinate measurement machines (CMM), theodolites, laser interferometers and vision systems.

Most existing calibration methods are expensive as they employ the use of expensive equipment. A CMM can cost in excess of US $100,000 depending on the required level of accuracy. Other methods require multiple laser interferometers and/or laser theodolites, which can cost US $100,000 each, and the use of precision machined fixtures, which typically require the use of a CMM as well.

In the last decade, a variety of proposed calibration methods have been offered as alternatives. Many different calibration methods are given an excellent review by Hollerbach & Wampler (1996). Explicit robot calibration methods differ according to measurement system. These are *open-loop methods* (requires an external system to measure the pose of the robot end-effector) and *closed-loop methods* (attaching the end-effector to ground, calibration is achieved using joint angle sensing only). Wampler, et al. (1995) present what they call *an implicit loop method*. They give a framework for treating the open-loop method as closed: the loop is closed through an external sensor.

One approach constrains the end-effector motion. Goswami et al. (1993) present a representative example of a constraint method that uses a simple radial-distance linear variable differential transformer (LVDT) that measures the distance from several fixed points in the workspace to the robot's end-effector. They employ the use of an accurate, inexpensive telescopic ball-bar system as a measuring device, where the ball-bar has a magnetic chuck permanently mounted on one end and a removable precision steel sphere mounted on the opposite end. The removable sphere allows the addition of extension rods to increase the reach of the robot's workspace. Constraint methods, such as this one, have the disadvantage of limiting the calibration workspace and require the addition of more mechanical pieces to increase the robot's reach. Additional pieces or larger fixturing devices will introduce more errors inherently (geometric and non-geometric) and will increase the cost. In the ball-bar method, there are also additional time and space requirements for setting up the fixturing system, which may not be ideal in a manufacturing location. Meanwhile, the removal of the constraint can cause the robot to drift and the calibration may become invalid (Wampler et al., 1995). This method is not task-specific, meaning the robot is calibrated over an environment that the robot may not be operating in or with the proper tool, which may introduce additional compliance errors, as would calibration with the addition of the fixture to the end-effector.

There has been much work in the use of laser systems for calibration. Vincze et al. (1994) present a comprehensive overview of existing laser solutions. A laser tracking system (LTS) can determine both the position and orientation of a robot's end-effector with high accuracy during arbitrary robot motions. However, a fixed laser beam method is limited as it only measures along a narrow straight line and requires the use of a heavy sensor head at the robot end-effector. Using three to four laser beams allows very precise length measurements with an interferometer to calculate position, but the orientation range is limited and the dynamic measurements can be inaccurate. Moreover, the line of sight between laser and interferometer must be maintained. In the case of a two-beam system, there is a requirement for two tracking mechanisms with two sets of angular encoder systems. Additionally, the two beams will have to be reflected, and the working range is more limited than for a one-beam system.

In a one-beam system, it is also possible to measure orientation, but only one precise mechanism for the deflection of the laser beam needs to be built, meaning less likelihood of link interference. Also, there is a decreased requirement on the orientation of the robot holding the reflector when the same reflector is used. The second deflection unit is replaced in the one-beam solution with an interferometer that provides excellent accuracy, but requirs an absolute measurement system or a calibration at start-up.

Vincze et al. (1994) present a one-beam solution that uses a steerable laser interferometer with a steerable reflector. This solution uses pitch and yaw measurements, with the steerable interferometer yielding all 3 components of position and the steerable reflector yielding two components of orientation. However, this solution uses a HeNe laser-interferometer, mirrors, a high precision cardan joint, retroreflectors, a position sensitive diode (PSD), and a CCD camera in an elaborate set-up. This system would not be easily implemented, or maintained on a manufacturing floor. Furthermore, HeNe lasers tend to be more fragile than solid-state lasers and may be easily damaged in a manufacturing environment. Depending on the laser class required by a particular solution, safety may also be a concern. The complexity of the equipment drives up the expense of this type of solution to the calibration problem.

Gong et al. (2000) present a self-calibrating system that can be easily implemented on a production line or manufacturing floor that does not require the use of a special measurement system. Their solution consists of a hybrid non-contact optical sensor (a combination of a CCD camera and a structured light sensor) that is mounted on the robot hand. The system is moved to measure some external targets where the distance between any two targets has previously been measured using a CMM. By comparing the distance measured by the robot system with the CMM measured distance, it is possible to calibrate the robot. The requirement for a CMM makes this solution costly. Also, their method requires a hand-to-sensor calibration to find the position and orientation of the sensor coordinate system with respect to the robot hand coordinate system, meaning the robot calibration is now limited by the hand-to-sensor calibration. An approach presented by van Albada, et al. (1995) has a similar dependence on a CMM.

The hand-to-sensor calibration adds complexity and lengthens the error chain. See Chang & Liang (1989), for example. The camera itself must be calibrated, but needs to be done only once. There are many approaches, those offered by Lenz & Tsai (1987) and Stein (1993) are representative of algebraic and geometric methods, respectively. We propose to use the camera calibration procedure put forward by Ofner (2000), which determined the unique projective transformation that straightens image lines warped by the linear portion of the distortion inherent to the camera-lens combination.

The kinematic calibration measurement concept presented in this paper uses a robot-mounted CCD camera and laser distance sensors, together with ruled and flat standards

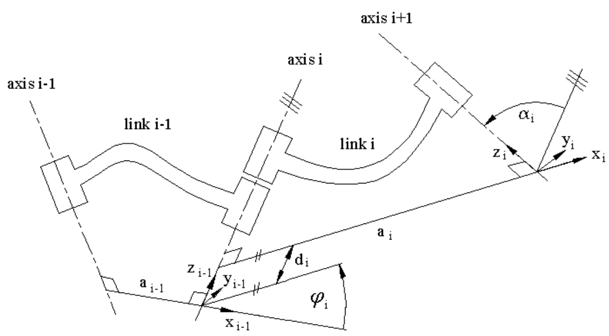

Figure 2. The DH parameters.

allowing for a low-cost, self-calibrating system that does not require external measurement and has the potential of achieving the same, or better accuracy of those that do.

Method Overview: The measurement concept and calibration method presented here uses a CCD camera and two laser distance sensors to measure a change in position and compare it to the commanded position change relative to ruled and flat standards. The difference in commanded and measured motions is used to iteratively adjust the Denavit-Hartenburg (DH) parameters, allowing the robot to correct for inaccuracies. The parameters are assigned according to Denavit and Hartenberg (1955), as shown in Figures 2 and 3. The new parameters are then used to accurately position the robot. A benefit of this method is that there is no need to distinguish between the geometric and non-geometric errors, which intrinsically involve dealing with non-linear parameters that are difficult to correctly analyze. Regardless, effects of geometric errors tend to be larger (about 95% of the overall error) than the effects of non-geometric errors (Gong et al., 2000(b)).

Another benefit of this method of calibration is that it is task-specific: the robot does not need to be calibrated over the entire workspace, only in the robot's task-space for that specific operation. Thus, task-specific calibration over small areas can lower the

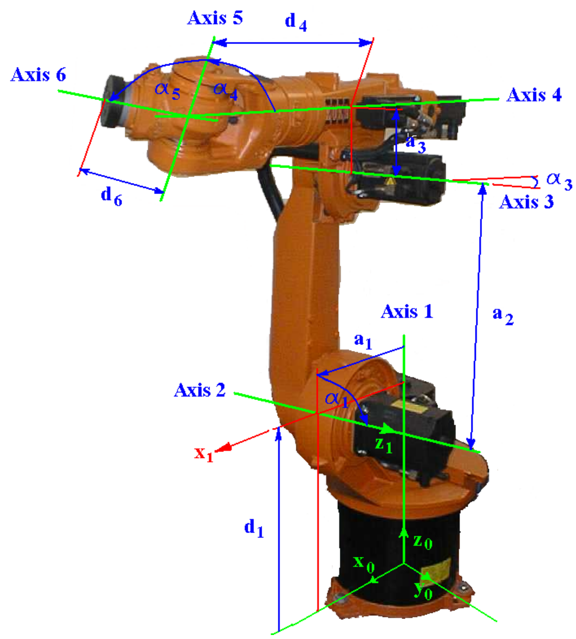

Figure 3. KUKA KR-15/2 DH Parameters.

amount of geometric errors that arise during the specific task. Compliance errors fall in the category of non-geometric errors and deal with the link deflection and flexibility of the robot joints when external loads are added, or with the weight of the robot (Gong et al., 2000(b)). The method discussed in this paper allows the robot to be calibrated with the addition of any external payload, including tools, resulting in lower compliance errors.

Finally, the proposed method only requires measurement of relative motions rather than the absolute end-effector positions in the workspace. This eliminates the need for a hand-to-sensor calibration, in turn shortening the error chain.

Kinematic Calibration Details: The forward kinematics of an n-link manipulator is give by:

$$^{\mathrm{b}}\mathbf{x} = \mathbf{f}(\boldsymbol{\rho}), \quad (1)$$

where $^{\mathrm{b}}\mathbf{x}$ is a 6 x 1 vector containing the 6 degree-of-freedom (DOF) Cartesian coordinates of the end-effector reference frame with respect to the base frame, $\boldsymbol{\rho}$ is the $4n \times 1$ vector of DH-parameters describing the kinematic geometry of the joint axes and $\mathbf{f}(\boldsymbol{\rho})$ is the kinematic transformation from the end-effector of the robot to the base.

In our model there are four DH parameters for each i of n-links in the manipulator: the joint angles, φ_i; the joint twist angles, α_i; the link lengths, a_i; and the link offsets, d_i. This accounts for the $4n \times 1$ in the parameter vector $\boldsymbol{\rho}$. The model can be made more comprehensive by adding parameters to account for other geometric, and non-geometric sources of error.

In the current calibration method, only the 3 DOF position coordinates of ${}^{\mathrm{b}}\mathbf{x}$ are required. The pose of a robot is specified by six independent position and orientation parameters. Because of this independence, subsets of these parameters can be used to formulate the system of calibration equations. Thus, it is not essential to use orientation measurements. Eliminating the end-effector orientation simplifies both the mathematics and the measurement process. The latter is important in order to keep the calibration inexpensive.

With the removal of the end-effector orientation, ${}^{\mathrm{b}}\mathbf{x}$ simplifies to a 3 x 1 position vector and $\mathbf{f}(\boldsymbol{\rho})$ reduces to the position vector component of the tip-to-base transformation matrix, ${}^{\mathrm{b}}\mathbf{T}_{\mathrm{t}}$. A small error in the end-effector position can be approximated as

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\rho}} \begin{bmatrix} \Delta \boldsymbol{\varphi} \\ \Delta \boldsymbol{\alpha} \\ \Delta \mathbf{a} \\ \Delta \mathbf{d} \end{bmatrix}, \tag{2}$$

or

$$\Delta \mathbf{x} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\rho}} \Delta \boldsymbol{\rho} = \mathbf{J} \Delta \boldsymbol{\rho}. \tag{3}$$

In Equation 3, $\Delta \mathbf{x}$, refers to the difference between where the robot thinks it is, via the nominal DH parameters and kinematic chain, and where it actually is. The vector $\Delta \boldsymbol{\rho}$ represents the corresponding errors in the DH-parameters. It is these DH-parameter errors that are to be calibrated out.

The matrix $\mathbf{J}$ is the full position Jacobian for the manipulator, i.e., a $3 \times 4n$ matrix of partial derivatives of $\mathbf{f}(\boldsymbol{\rho})$ with respect to each of the $4n$ DH-parameters:

$$\mathbf{J} \equiv \left[\begin{matrix} \frac{\partial f_1}{\partial \phi_1} \cdots \frac{\partial f_1}{\partial \phi_6} & \frac{\partial f_1}{\partial \alpha_1} \cdots \frac{\partial f_1}{\partial \alpha_6} \\ \frac{\partial f_2}{\partial \phi_1} \cdots \frac{\partial f_2}{\partial \phi_6} & \frac{\partial f_2}{\partial \alpha_1} \cdots \frac{\partial f_2}{\partial \alpha_6} \\ \frac{\partial f_3}{\partial \phi_1} \cdots \frac{\partial f_3}{\partial \phi_6} & \frac{\partial f_3}{\partial \alpha_1} \cdots \frac{\partial f_3}{\partial \alpha_6} \end{matrix} \right. \left. \begin{matrix} \frac{\partial f_1}{\partial a_1} \cdots \frac{\partial f_1}{\partial a_6} & \frac{\partial f_1}{\partial d_1} \cdots \frac{\partial f_1}{\partial d_6} \\ \frac{\partial f_2}{\partial a_1} \cdots \frac{\partial f_2}{\partial a_6} & \frac{\partial f_2}{\partial d_1} \cdots \frac{\partial f_2}{\partial d_6} \\ \frac{\partial f_3}{\partial a_1} \cdots \frac{\partial f_3}{\partial a_6} & \frac{\partial f_3}{\partial d_1} \cdots \frac{\partial f_3}{\partial d_6} \end{matrix} \right]. \tag{4}$$

Usually, the Jacobian is calculated under the assumption that all the DH-parameters except the joint variables remain constant, making it a $3 \times n$ matrix (or $6 \times n$ if the orientation is included). That is not the case here where small errors are assumed in all DH-parameters. Since the Jacobian in Eq. (4) is generally non-square, it is not invertible, so the solution to Eq. (3) requires the pseudo-inverse of $\mathbf{J}$. The Moore-Penrose formulation yields

$$\Delta \boldsymbol{\rho} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \Delta \mathbf{x} = \mathbf{J}^{+} \Delta \mathbf{x}. \tag{5}$$

The general calibration procedure is to measure the end-effector error, $\Delta \mathbf{x}$, and calculate $\Delta \boldsymbol{\rho}$ using Eq. 5. Since there are $4n$ unknowns in $\Delta \boldsymbol{\rho}$ and only three values in $\Delta \mathbf{x}$, at least $4n/3$ poses are required for a complete determination of the DH-parameter errors. Each pose creates a set of three equations via Eq. 3, which can be stacked for m-poses:

$$\begin{bmatrix} \Delta \mathbf{x}_1 \\ \Delta \mathbf{x}_2 \\ \Delta \mathbf{x}_3 \\ \vdots \\ \Delta \mathbf{x}_m \end{bmatrix} = \begin{bmatrix} \mathbf{J}_1(\boldsymbol{\rho}_1) \\ \mathbf{J}_2(\boldsymbol{\rho}_2) \\ \mathbf{J}_3(\boldsymbol{\rho}_3) \\ \vdots \\ \mathbf{J}_m(\boldsymbol{\rho}_m) \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{\phi} \\ \Delta \boldsymbol{\alpha} \\ \Delta \mathbf{a} \\ \Delta \mathbf{d} \end{bmatrix}. \tag{6}$$

If exactly $4n/3$ poses are measured, Eq. (5) will provide an estimate of $\Delta \boldsymbol{\rho}$. If more than

$4n/3$ poses are measured, $\Delta\boldsymbol{\rho}$ in Eq. (5) can be fit, in a least-squares sense, to the measured data.

A problem in the calculation of $\mathbf{J}^+$ in Eq. (5) exists if $\mathbf{J}$ is singular or numerically nearly singular. This is addressed using a singular value decomposition (SVD) of $\mathbf{J}$,

$$\mathbf{J} = \mathbf{USV}^T, \tag{7}$$

where $\mathbf{U}$ is a $3 \times 4n$ matrix with orthogonal columns, $\mathbf{S}$ is a $4n \times 4n$ diagonal matrix whose elements are the singular values of $\mathbf{J}$, and $\mathbf{V}$ is a $4n \times 4n$ orthonormal matrix. The pseudo-inverse of $\mathbf{J}$ is then,

$$\mathbf{J}^+ = \mathbf{VS}^{-1}\mathbf{U}^T. \tag{8}$$

Since $\mathbf{S}$ is a diagonal matrix, $\mathbf{S}^{-1}$ is also a diagonal matrix whose elements are $1/s_i$. If any of the singular values of $\mathbf{S}$ are zero, then the corresponding values of $\mathbf{S}^{-1}$ are set to zero, rather than ∞. This is also applied to values of $\mathbf{S}$ that are nearly zero. For the sake of numerical calculation, "nearly zero" refers to singular values s_i whose ratio to the maximum singular value is less than rank(V) (i.e. the dimension of the column space of V) times the machine precision, ϵ, $s_i/s_{\max} \leq \text{rank}(V)\epsilon$, as recommended by Press, et al. (1992) in *Numerical Recipes in C*.

Setting the nearly zero values to zero removes information from $\mathbf{J}$, which appears to conflict with the goals of performing calibration. However, values near zero are prone to numerical processing errors and indicate DH-parameters that are close to being unobservable in the given robot configuration. (Singular values that are zero indicate the corresponding DH-parameters are not observable at all). What this means is that small variations in $\Delta\mathbf{x}$ could result in large changes in the calculation of $\Delta\boldsymbol{\rho}$. The information yielded by the very small singular values should be thrown out because it is, at best, useless. Bad information is worse than no information.

The calibration proceeds by iterating Eq. (5) with the updated DH-parameters,

$$\boldsymbol{\rho}_i = \boldsymbol{\rho}_{i-1} + \Delta\boldsymbol{\rho}, \tag{9}$$

where i indicates the iteration step. (Note that $\boldsymbol{\rho}_0$ is required for the first iteration, which refers to the nominal DH-parameter set.) The location of where the robot thinks it is can be found by substituting Eq. (9) into Eq. (1), thereby computing the forward kinematics:

$$\mathbf{x}_{i(\text{comp})} = \mathbf{f}(\boldsymbol{\rho}_i) = \mathbf{f}(\boldsymbol{\rho}_{i-1} + \Delta\boldsymbol{\rho}). \tag{10}$$

In an absolute measurement based system, each pose, $\mathbf{x}_{\text{meas}}$, is measured only once, and remains constant while $\mathbf{x}_{i(\text{comp})}$ is updated with each iteration. In the relative measurement based system presented herein, the measurement data directly yields values of $\Delta\mathbf{x}$. In absolute measurement based systems the $\Delta\mathbf{x}$ in Eq. (6) are computed as,

$$\Delta\mathbf{x}_i = \mathbf{x}_{\text{meas}} - \mathbf{x}_{\mathbf{i}(\text{comp})}. \tag{11}$$

Knowing $\Delta\mathbf{x}$ from the processed data provided by the vision and laser measurement system, and knowing the initial value of $\mathbf{x}_{\text{comp}}$ means that the absolutely measured pose is embedded in the existing relatively measured data. But, for estimating the DH parameter correction factors, $\Delta\boldsymbol{\rho}$, we require $\mathbf{x}_{\text{meas}}$ so that we can iteratively update $\Delta\boldsymbol{\rho}_i$ until it is vanishingly small. Hence, after the relative measurement data, $\Delta\mathbf{x}$, is acquired and the initial $\mathbf{x}_{\text{comp}}$, based on the nominal DH parameters $\boldsymbol{\rho}_0$, is computed, the absolute measurement data may be extracted from Eq. (11) using initial values:

$$\mathbf{x}_{\text{meas}} = \Delta\mathbf{x}_0 + \mathbf{x}_{0(\text{comp})}. \tag{12}$$

Now, Eq. (5) can be solved for the DH parameter errors using SVD. Each subsequent iteration provides a new estimation of $\Delta\boldsymbol{\rho}_i$ until the convergence criterion, κ, in Eq. (13) is satisfied:

$$\| \mathbf{J}\Delta\boldsymbol{\rho} - \Delta\mathbf{x} \| \leq \kappa. \tag{13}$$

It is important to emphasize that it is not necessary to measure the absolute position of the robot, $\mathbf{x}_{meas}$. It is sufficient to measure the difference between actual (measured) and computed positions between two poses, $\Delta\mathbf{x}$.

Combining all of the above details, the parameter identification algorithm can be summarized in the following nine steps:

1. Move the robot through $m+1$ poses. Compute the initial $\mathbf{x}_{comp}$ using the nominal DH parameters and the forward kinematics transformation from Eq. (10). After the measurement data has been processed, compute $\mathbf{x}_{\text{meas}}$ using Eq. (12). Each element in the vector $\mathbf{x}_{\text{meas}}$ is determined from the difference between two adjacent poses, hence for a vector of dimension m, $m+1$ differences are required. The values contained in $\mathbf{x}_{\text{meas}}$ are assumed to be constant and need only be computed once at the start of the identification algorithm.
2. Compute the Jacobian, $\mathbf{J}$ based on the nominal DH parameters.
3. Now the iterative parameter identification commences. The first step is to compute $\Delta\boldsymbol{\rho}$ using Eq. (5) and SVD.
4. Check if the convergence criterion,

   $$\parallel \mathbf{J}\Delta\boldsymbol{\rho} - \Delta\mathbf{x} \parallel \leq \kappa$$

   has been satisfied. If not, continue.
5. Update the DH parameters using Eq. (9)
6. Update the Jacobian, $\mathbf{J}$, using Eq. (4).
7. Update $\mathbf{x}_{comp}$ using Eq. (10).
8. Update $\Delta\mathbf{x}$ using Eq. (11).
9. Go to step 3.

Simulation: A MATLAB simulation of a calibration experiment was performed as a *proof of concept*. The parameter identification algorithm was run using synthetically generated errors and measurements. That is, a fictitious set of DH-parameter errors were created,

$$\boldsymbol{\rho}_{\text{actual}} \quad = \quad \boldsymbol{\rho}_{\text{nominal}} + \Delta\boldsymbol{\rho}_{\text{synthetic}}.$$

A perfect measurement system simulator then provided

$$\Delta\mathbf{x} \quad = \quad \mathbf{x}_{\text{meas}} - \mathbf{x}_{\text{comp}}.$$

A KUKA KR-15/2 robot was modelled using DH parameters (see Table I). An arbitrary home position was selected and constant increments were sequentially added to the home joint angles to change the robot configuration, these are all listed in Table II. The increments were selected so the final position would be in the workspace after 100 increment steps, and so that all the joints would move in different ways. The number of measured positions was set to be 100. The tolerance on the smallness of singular values was set to be $\text{rank}(V)\epsilon \approx 5.33 \times 10^{-15}$ ($\text{rank}(V) = 24$ and on the computer used $\epsilon = 2.2204 \times 10^{-16}$), and the convergence criterion was set to be $\kappa = 10^{-8}$, a value arrived at by trial and error. The selection of $\kappa = 10^{-8}$ is justified by the results below. Computed poses were determined with the nominal DH parameters and measured poses were determined with some assigned parameter errors. The calibration procedure was then run.

i	φ_i	α_i (deg.)	a_i (m)	d_i (m)
1	φ_1	90	0.300	0.675
2	φ_2	0	0.650	0
3	φ_3	90	0.155	0
4	φ_4	-90	0	0.600
5	φ_5	90	0	0
6	φ_6	0	0	0.140

Table I
DH parameter assignments.

Initial position (deg.)		Increments (deg.)	
φ_1	0.0	$\Delta\varphi_1$	-3.0
φ_2	-90.0	$\Delta\varphi_2$	3.0
φ_3	0.0	$\Delta\varphi_3$	-2.0
φ_4	0.0	$\Delta\varphi_4$	-3.5
φ_5	0.0	$\Delta\varphi_5$	3.2
φ_6	0.0	$\Delta\varphi_6$	-2.5

Table II
Initial joint angles; constant increments.

Parameter error	Synthetic	Identified	% Difference
Joint Angle	(rad)	(rad)	(%)
$\Delta\varphi_1$	0.000870	0.000870	0
$\Delta\varphi_2$	0.000940	0.000940	0
$\Delta\varphi_3$	-0.001000	-0.001000	0
$\Delta\varphi_4$	0.000620	0.000620	0
$\Delta\varphi_5$	-0.000810	-0.000807	0.37
$\Delta\varphi_6$	0.000260	0.000000	100.00
Twist Angle	(rad)	(rad)	(%)
$\Delta\alpha_1$	0.000157	0.000157	0
$\Delta\alpha_2$	0.000130	0.000130	0
$\Delta\alpha_3$	-0.000160	-0.000160	0
$\Delta\alpha_4$	-0.000253	-0.000253	0
$\Delta\alpha_5$	0.000462	0.000464	-.41
$\Delta\alpha_6$	-0.000320	-0.000320	0
Link Length	(m)	(m)	(%)
Δa_1	0.000031	0.000031	0
Δa_2	0.000051	0.000051	0
Δa_3	0.000012	0.000012	0
Δa_4	-0.000045	-0.000045	0
Δa_5	0.000064	0.000063	1.60
Δa_6	0.000058	0.000058	0
Link Offset	(m)	(m)	(%)
Δd_1	-0.000075	-0.000075	0
Δd_2	0.000031	-0.000069	322.05
Δd_3	0.000022	0.000122	-453.80
Δd_4	0.000048	0.000048	0
Δd_5	-0.000020	-0.000019	3.22
Δd_6	0.000078	0.000078	0

Table III
Synthetic, identified parameter errors, % difference for simulated calibration experiment.

Results: The MATLAB routines simulating the calibration experiment converged after two iterations and a total CPU time of 46.48 seconds, reporting that

$$\| \mathbf{J}\Delta\boldsymbol{\rho} - \Delta\mathbf{x} \| = 3.5697e - 013.$$

We believe maintaining the norm three orders of magnitude larger than ϵ to be prudent. Values of κ need to be smaller than 10^{-13} to improve the parameter estimation. This is in a range where round-off error could bias results, which requires closer investigation in future work. Table III lists the results together with the % difference relative to synthetic parameter error values (note that degrees are given in *rads*, while lengths are in m). The calibration procedure identified 17 of the 24 parameter errors exactly. Four additional parameters, $\Delta\varphi_5$, $\Delta\alpha_5$, Δa_5 and Δd_5 were identified with less than 4% difference from the assigned synthetic error value, moreover the signs of these four errors agree with the assigned ones. It is interesting to note that all four of these parameter errors relate to the fifth robot joint axis. Three parameter errors, $\Delta\varphi_6$, Δd_2, and Δd_3 were, for all intents and purposes, not identified.

The SVD algorithm annihilated one singular value, indicating that one parameter error was unobservable for this set of synthetic measurements. The two next smallest singular values were 4 and 1 orders of magnitude smaller than the third smallest. To address this issue requires investigation of the effects of the SVD singular value criteria on identification.

The question remains, how does one know which of the parameters have been accurately identified in the absence of apriori knowledge? The answer, at least partially, will likely be found by closer examination of the statistical variance of the SVD output. It turns out that the parameter variance is a by-product of the SVD.

Conclusions and Future Work: The calibration method presented in this paper leads to a low-cost (US $7,000 - US $10,000) system constructed with readily available off-the-shelf items. Furthermore, it is straightforward to implement on a production line or manufacturing floor and can easily be applied to any serially connected robotic manipulator. The system is also task-specific, meaning a robot can be calibrated with the appropriate tool and over the task-space being used, as opposed to the entire workspace, thereby correcting geometric errors and improving position accuracy without the need to absolutely measure arbitrary poses covering the reachable workspace of the robot.

Results of a MATLAB simulation using synthetically generated data indicate that the measurement concept and parameter identification procedure provides accurate corrections to the DH-parameters. The calibration was also successfully applied to a real set of test measurements on a Kuka KR-15/2 robot, although the accuracy of the identified parameters have not yet been experimentally verified. A new simulation imposing linear robot motions is being developed.

The corrected DH-parameters should be tested using the same positioning to quantify the improvement in the robot's accuracy and to calculate any remaining errors. Also, measurements should be taken to compare the simulation to the actual robot movement to see if there is a difference in the simulated parameters. Additional investigation into the statistical confidence intervals provided by the SVD algorithm is necessary for the development of control algorithms for automating the calibration process. This would provide the basis for the calibration system to decide which identified parameters are likely to be trustworthy.

It would be worthwhile to check this method with the addition of a tool (payload). This would give an indicate of how compliance errors may contribute to robot manipulator inaccuracy. Because the system is task-specific, it must provide for calibration for the task at hand, including the addition of a payload, crucial for pick-and-place.

Ideally, the updated DH-parameters would be fed back into the robot controller as they are calculated, making the calibration system operational in "real-time". At this time, it is not possible to do this because commercial robot control architecture is closed; instead, these new parameters need to be used to alter the desired position (trajectory) to complete the calibration. A method of feeding the new parameters back into the robot control system would complete the calibration system and is something to address in the future. This would likely involve working closely with a robot manufacturer to design software and hardware to interface with the control system that will directly feed in the parameter changes, as determined for that task.

References

[1] Chang, Y., and Liang, P., "On recursive calibration of cameras for robot hand-eye systems", Proceedings of the 1989 IEEE International Conference on Robotics and Automation, Arizona, pp. 838-843, 1989.

[2] Denavit, J., Hartenberg, R.S., "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices", J. of Applied Mechanics, pp. 215-221, 1955.

[3] Gong, C., Yuan, J., Ni, J., "A Self-Calibration Method for Robotic Measurement System", Journal of Manufacturing Science and Engineering, Transactions of the ASME, Vol. 122, pp. 174-181, 2000.

[4] Gong, C., Yuan, J., and Ni, J., "Nongeometric error identification and compensation for robotic system by inverse calibration", International Journal of Machine Tools & Manufacture, Vol. 40, pp. 2119-2137, 2000(b).

[5] Goswami, A., Quaid, A., and Peshkin, M., "Complete parameter identification of a robot pose from partial pose information" IEEE Intl. Conf. Robotics and Automation. Atlanta, pp. 168-173, May 2-7, 1993.

[6] Hollerbach, J.M., and Wampler, C.W., "The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods", The International Journal of Robotics Research, Vol. 15, 1996.

[7] Lenz, R.K., and Tsai, R.Y., "Techniques for calibration of the scale factor and image centre for high accuracy 3D machine vision metrology", Proceedings of the 1987 IEEE International Conference on Robotics and Automation, Raleigh, N.C., March 31 - April 3, pp. 68-75, 1987.

[8] Mooring, B.W., Roth, Z.S., and Driels, M.R., "Fundamentals of Manipulator Calibration", John Wiley & Sons, 1991.

[9] Ofner, R., "Three-Dimensional Measurement via the Light-Sectioning Method", Ph.D. Thesis, Institute for Automation, University of Leoben, Leoben, Austria, 2000.

[10] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P., "Numerical Recipes in C: The Art of Scientific Computing", Second Edition, Cambridge University Press, 1992.

[11] Stein, G.P., "Internal Camera Calibration using Rotation and Geometric Shapes", M.Sc. thesis, Dept. of Electrical Engineering, MIT, 1993.

[12] Wampler, C.W., Hollerbach, J.M., and Arai, T., "An Implicit Loop Method for Kinematic Calibration and Its Application to Closed-Chain Mechanisms", IEEE Transactions on Robotics and Automation, Vol. 11, No. 5, October 1995.

[13] van Albada, G.D., Lagerber, J.M., Visser, A., and Hertzberger, L.O., "A low-cost pose-measuring system for robot calibration", Robotics and Autonomous Systems, Vol. 15, pp. 207-227, 1995.

[14] Vincze, M., Prenninger, J.P., and Gander, H., "A Laser Tracking System to Measure Position and Orientation of Robot End Effectors Under Motion", International Journal of Robotics Research, Vol. 13, No. 4, pp. 305-314, August 1994.