Homography-based Trajectory Tracking of Industrial Manipulator with Iterative Learning Control

Bingxi Jia, Shan Liu, Yi Liu
Department of Control Science and Engineering, Zhejiang University, Hangzhou, China
bxjia@zju.edu.cn, sliu@iipc.zju.edu.cn, ly30902426@zju.edu.cn

Abstract—In this paper, a repetitive assembling task is considered for industrial manipulator, where vision-based precise trajectory tracking is necessary. Conventional visual servoing methods suffer from their poor dynamic performance, while the proposed open-plus-closed loop framework solved this problem by iterative learning. System error is defined by the directly measurable homography matrix and feature points, without any homography decomposition. Besides, in the proposed controller, image Jacobean is not used so that no singularity problem exists. System convergence is analyzed and simulation results are given to demonstrate the effectiveness, i.e. fairly accurate trajectory tracking is achieved.

Keywords—Homography-based; trajectory tracking; iterative learning control

I. INTRODUCTION

Industrial manipulators are widely used in many areas such as part assembling, welding, etc. However, most of them are using teaching-playing strategy, i.e. a human operates the manipulator first to accomplish the task and the trajectory of joints are recorded to play again and again by the robot to perform a repetitive task. Such a system is not flexible and needs much effort for engineers to operate the robot and program. To this end, vision-based methods are proposed a few decades ago[1], vision-based control methods use visual feedback instead of commonly-used physical sensor signals in the control loop, achieving good flexibility to perform complicated tasks.

Vision-based control methods can be divided into three kinds, i.e. position-based, image-based, and hybrid methods. Recent reviews are [2, 3]. By contrast, image-based and hybrid methods are more robust to environment uncertainties and rely less on the accuracy of the vision system[4]. Generally, for image-based methods only monocular camera is used without the need of the expensive and carefully calibrated stereo vision system. However, due to the loss of depth information in the projection of points, model uncertainties exist in both image-based and hybrid methods and affect the stability and convergence of control system. Lots of adaptive control methods are proposed to deal with the model uncertainties [5-7], online adaptive parameter identification is used to guarantee the stability of the control loop. Besides, some robust control methods are also proposed such as sliding mode control[8], mixed control[9], etc. Most of existing methods are designed for set-point control task, and suffer weak dynamic performance when performing a trajectory tracking task.

Iterative learning control is widely used for industrial repetitive tasks, and precise trajectory tracking can be achieved. In [10, 11] iterative learning control is used for visual trajectory tracking where eye-to-hand system is considered. In [10] a filter-based iterative learning method is proposed for a general nonlinear model, image feature velocity is used as the generalized input for iteration and robot velocity is computed using the inverse of image Jacobean. Thus the system may suffer from the singularity of Jacobean and the proposed method can only be used for fixed camera configuration. In [11] an indirect iterative learning method is proposed for a discrete visual servoing task, artificial neural network(ANN) model is used to approximate the local camera-robot model at every sample point, and iterative learning method is used for ANN identification. A method to avoid singularity is also considered in that paper, but the system is computationally complex.

As in Fig.1, a 6-DOF manipulator equipped with a monocular camera on the tip is considered in this paper. The task is to assemble the handling object with the target using visual feedback. First, a human teacher grasps it with a camera on it to assemble with the target as a demonstration, a set of images are captured in the process and are considered as the desired trajectory. The manipulator then grasps it and performs repetitive tracking to accomplish the assembling task. Homography is used for trajectory definition and tracking, compared with feature point based methods, it is more robust to image noise owing to the effective homography estimation methods[12], and at least 3 points are enough to determine homography thus it is capable of dealing with partial target loss. Compared with conventional methods in which homography decomposition is used[5], the error vector here is defined directly based on homography and image measurement, thus is more time-saving and robust to parameter
uncertainties. Besides, an open-plus-closed-loop control strategy is proposed which is model-free, thus no singularity problem exists and precise trajectory tracking can be achieved owing to the iterative learning method.

II. PROBLEM FORMULATION

As in Fig.1, Consider a 6-DOF robot manipulator equipped with a monocular camera on the tip, the task is performed using teach-by-doing strategy. The target object is expressed as a set of coplanar points \( \{ P_i \} \) that can constraint the pose. First the human teacher performs the task for demonstration and a series of images \( \{ I \} \) are captured that express the desired trajectory, where \( j \) denotes the \( j \)-th sampling. The image captured at the desired pose is \( \ast I \). \( \{ P_i \} \) are projected onto \( \ast I \) as image points \( \{ p_i \} \), and the corresponding points are \( \{ \ast p_i \} \) at the desired pose. The following assumptions are made for practical use:

1) \( \{ P_i \} \) are visible at the initial pose and always in camera’s field-of-view in the teaching process;
2) The trajectory in workspace is smooth and differentiable;

After the teaching process, the manipulator tracks the recorded trajectory iteratively.

III. MODELING

In this section, homography-based model is given to express the system, compared to those that directly use point-like features, the proposed homography-based method is more robust to the image noise and partial target loss. In this paper, the desired trajectory is defined by the homography with respect to the target frame, and the error signal is constructed directly on the measurable homography matrices and image coordinates, without any metric information. What’s more, as the trajectory is not defined directly by the image features, the system works well even if the appearance of the target object changes, improving its reliability in industrial environment.

A. Homography

As in Fig.2, consider a set of coplanar points \( \{ P_i \} \), let \( \chi_a = [X_a \ Y_a \ Z_a]^T, \chi_b = [X_b \ Y_b \ Z_b]^T \) denote the coordinates with respect to frame \( a,b \), and \( (R,t) \) denotes the relative pose between the two frames, the following relationship can be determined:

\[
\chi_b = R\chi_a + t
\]

Let \( n \) be the unit normal vector of the plane expressed in \( C_a \), and \( d \) is the distance from the origin of \( C_a \) to the plane along \( n \), which is given by:

\[
d = n^T \chi_a
\]

Using(2), the expression given in (1) can be rewritten as:

\[
\chi_b = H\chi_a
\]

where the Euclidean homography \( H \) is defined as:

\[
H = R + t n^T d
\]

Let \( m_a, m_b \) be the normalized coordinates, which are given by:

\[
m_a = \frac{1}{Z_a} \chi_a, m_b = \frac{1}{Z_b} \chi_b
\]

Then they are projected on the image by the following expression:

\[
p_a = Km_a, p_b = Km_b
\]

From (3)(5) it can be inferred that their projection onto image plane at two different poses are related by a homography[13]:

\[
\frac{Z_b}{Z_a} m_a = Hm_b
\]

Homography has been widely used in visual servo control, the conventional method is to decompose the homography matrix to get pose parameters \((R,t,n)\) when \( K \) is known[14]. In general, four solutions exist and two of them are physically possible. An approximation of the real normal vector \( n \) is necessary to make the choice. These methods achieved good performance while may suffer from the inaccurate camera parameters and image noise.
B. Error System

As in Fig. 3, at the time $k$, the desired relative pose with respect to the target frame is $T_r$ and the current $T_k$, thus the error vector at time $k$ can be constructed by the relative pose with respect to the desired frame:

$$T_e = T_k \cdot T_r^{-1} \tag{8}$$

where $T = \begin{bmatrix} R & t \\ 0_{1x3} & 1 \end{bmatrix}$ is the transform matrix.

Consider a point $P$ on the target object, it is projected onto the three frames and the corresponding coordinate as well as normalized coordinate are denoted as: $\{X', X, \hat{X}\}, \{m', m, \hat{m}\}$.

Denote $H_r$ as the homography between frame $F_r$ and $F^*$, denote $H$ as the homography between $F_k$ and $F^*$, and they are the measurable variables in the process, the following relation holds:

$$\begin{bmatrix} Z_r \end{bmatrix} m = H m^* \tag{9}$$

Thus the relationship between current and desired frame is given by:

$$\begin{bmatrix} Z_r \end{bmatrix} m = H H_r^{-1} m_r = H m_r \tag{10}$$

The error vector is divided into translational and rotational elements, which is:

$$e = \begin{bmatrix} e_t \\ e_r \end{bmatrix} \tag{11}$$

The translational element can be constructed by the difference between the coordinates in frames $F_r$ and $F_k$, which is given by:

$$d_r = \begin{bmatrix} (T_r - I) \cdot Z_r \\ 1 \end{bmatrix} \tag{12}$$

Substitute (5)(7) into(12), and neglect the constant elements we have:

$$e_i = \begin{bmatrix} Z_r \end{bmatrix} m - m_r = (H_r - I) m_r \tag{13}$$

Obviously $e_i$ can be computed only by the homography and image measurements, without the decomposition.

To construct the rotational error vector, we simply use $H_r - H^*$ which is anti-symmetric, thus the rotational error vector is given by:

$$e_r = \text{vex}(H_r - H^*) \tag{14}$$

It can be inferred from Theorem 1 in [15] that frame $F_k$ and $F_r$ coincide if and only if $e = 0$.

C. Trajectory Definition

As mentioned above, the desired trajectory of the manipulator in workspace can be expressed by the homography with respect to the target frame. When the teaching process finishes, a set of homography matrix $\{H_j\}$ is calculated based on the sampled images. In the control process, the trajectory is generated by interpolation. Note that in industrial workspace, the target object maybe covered by dust and its appearance changes gradually. Compared to trajectory definition directly based on image feature, homography-based definition is more reliable, i.e. the system doesn’t need to be re-taught even if the target image changes. Because in the error system, current target image and the taught target image is not necessarily the same, and the target image can be updated in the iterative process.

IV. CONTROL STRATEGY

A. System Architecture

As in Fig. 4, for the iterative trajectory tracking task, an open-plus-closed-loop (OPCL) framework is proposed. There exists a simple feedback controller in the closed loop for system stabilization and the iterative learning law uses previous process data for feed forward adjustment. The feedback controller improves the robustness of system and provides a good initial value for iterative learning. Precise tracking can be achieved owing to the iterative learning strategy.
B. Iterative Control Strategy

From the above mentioned, the desired trajectory is generated discretely, which is denoted as \( \{ H_i \} \). For controller design, the trajectory is interpolated to get a continuous trajectory \( H_i(t) \). Following controllers are based on the error vector \( e \) developed in (11). In iteration \( k \), the control input includes the feedback as well as feed forward parts:

\[
u_k(t) = u_{k,0}^u(t) + u_{k,1}^u(t)
\]

The feedback controller is designed as:

\[
u_{k,0}^u(t) = -K_p e(t)
\]

where \( \lambda \) is the controller gain which is a positive constant.

The iterative learning law is defined as:

\[
u_{k,1}^u(t) = u_{k-1}(t) + K_p e_{k-1}(t) + K_e e_{k-1}(t)
\]

where \( K_p, K_e \) are gains satisfying the stability condition.

C. Convergence Analysis

In this subsection, the convergence condition of the proposed iterative control strategy will be proved. Before representing the convergence proof, \( \lambda \)-norm definition, corresponding lemma and assumptions will be defined as follows.[16]

**Definition 1.** The \( \lambda \)-norm for a function \( h(t) \) is

\[
\|h\|_{\lambda} = \sup e^{-\lambda t} \|h\|, t \in [0, T]
\]

where \( \lambda \) is a positive scalar.

**Definition 2.** The error system is rewritten into the following normalized form:

\[
\dot{x} = f(x) \cdot u
\]

where \( x = [e, \omega] \) is system state, \( u = [v, \omega] \) is the control input. Similar to [15], \( f(x) \) is the system Jacobian which is given by:

\[
f(x) = \begin{bmatrix}
1/Z_x & -e_i + m_i \\
-u_i & -[u_i, [u_i]] + 2L_x
\end{bmatrix}
\]

If we rewrite the rotational matrix as \( R = e^{\theta \theta} \), where \( \theta \in [-\pi, \pi] \) is the rotation angle and \( u \) is the unit vector corresponding to the rotation axis, then \( L_x \) is given by:

\[
L_x = I - \frac{\sin \theta [u]}{2} - \sin^2 \frac{\theta}{2} \left( 2I + [u, [u]] \right)
\]

Parameters in (20) can be implicitly calculated from \( H \) by homography decomposition methods. \( Z_x, m_i \) are independent of \( x \) and bounded, \([u_i] \) and \([u_i] \) are constants. Therefore, it can be found that \( L_x \) satisfies the global Lipschitz continuity condition, and \( f(x) \) also satisfies the global Lipschitz continuity condition

\[
\|f(x(t)) - f(x_2(t))\| \leq k_j \|x_1(t) - x_2(t)\|
\]

In addition, \( f(x) \) is also bounded, \( \|f(x)\| \leq k_i \).

**Lemma 1.** Let \( \{ \alpha_k \} \) be a real sequence defined as

\[
\alpha_k = \rho \alpha_{k-j} + b_j, 0 \leq j \leq M - 1
\]

where \( b_j \) is a real sequence satisfying \( \lim_{k \to \infty} \sub b_k \leq b_\infty \).

If \( 0 < \rho < 1 \), then \( \lim_{k \to \infty} \sub a_k \leq \frac{b_\infty}{1-\rho} \).

**Assumption 1.** A continuous desired output \( x_d(t) \) for \( t \in [0, T] \) is available with a unique input \( u_d(t) \), and the desired control input \( u_d(t) \) is bounded, i.e., \( \|u_d(t)\| \leq b_d \) and \( b_d \) is a positive number.

**Assumption 2.** All operations start from the initial condition \( x_k(0) = x_0(0) \) for all \( k = 1, 2, \ldots, n \).

According to the aforementioned definition, lemma and assumptions, the sufficient convergence condition of the proposed iterative control strategy is reasoned and summarized in theorem 1.

**Theorem 1.** For error system, the iterative learning law (15) is applied iteratively under the aforementioned Assumptions. If the following inequality

\[
\|I - K_d f(x_k(t))\| \leq \rho_i < 1, t \in [0, T]
\]

holds for all \( x_k(t) \), then the output errors \( e_k(t) \) will converge to zero as \( k \to \infty \).

**Proof.** From (15)-(17), we can get that

\[
\Delta u_k(t) = \Delta u_{k-1}(t) - K_d(x_k(t) - x_{k-1}(t))
\]

\[
- K_d[(f(x_k(t)) - f(x_{k-1}(t)))]u_d(t) + f(x_{k-1}(t))\Delta u_{k-1}(t)]
\]

\[
+ K_p(x_k(t) - x_{k-1}(t))
\]

\[
= (I - K_d f(x_{k-1}(t)) - K_d(x_k(t) - x_{k-1}(t))
\]

\[
- K_d[(f(x_k(t)) - f(x_{k-1}(t)))]u_d(t)] + K_p(x_k(t) - x_{k-1}(t))
\]

Taking the norms on both sides of (24):

\[
\|\Delta u_k(t)\| \leq \|I - K_d f(x_{k-1}(t))\|\|\Delta u_{k-1}(t)\| + (k_p + k_d \|k_p\| \|x_{k-1}(t)\|)
\]

\[
+ k_p \|\Delta x_{k-1}(t)\| + k_p \|\Delta x_{k-1}(t)\| + k_d \|\Delta x_{k-1}(t)\|
\]

Moreover, we can also get
\[ \Delta x_{k+1}(t) = \Delta x_{k+1}(0) + \int_0^t f(x_\tau)u_\tau - f(x_{k+1})(\Delta u_{k+1})d\tau \]  
(26)

Taking the norms on both sides of (26):

\[ \|\Delta x_{k+1}\| \leq \int_0^t [k_1, k_2] \|\Delta x_{k+1}(\tau)\| + k_1 \|\Delta u_{k+1}(t)\|d\tau \]  
(27)

According to the Bellman-Cronwall Lemma[17], (27) can be rewritten into:

\[ \|\Delta x_{k+1}(t)\| \leq k_1 \int_0^t \|e^{b_2(t-\tau)}\|\Delta u_{k+1}(\tau)\|d\tau \]  
(28)

where \(b_2 = k_1, k_2\). Substituting (28) into (25) we obtain:

\[ \|\Delta u_k(t)\| \leq \rho \|\Delta u_{k-1}(t)\| + (k_p + k_2k_1k_2)k_1 \int_0^t \|e^{b_2(t-\tau)}\|\Delta u_{k+1}(\tau)\|d\tau \]  
(29)

Multiplying \(e^{-2t}\) on both sides of (29) to obtain:

\[ e^{-2t}\|\Delta u_k(t)\| \leq \rho e^{-2t}\|\Delta u_{k-1}(t)\| + (k_p + k_2k_1k_2)k_1 \int_0^t \|e^{b_2(t-\tau)}\|e^{-2\tau}\|\Delta u_{k+1}(\tau)\|d\tau \]  
(30)

To simplify (30) as:

\[ \|\Delta u_k(t)\| \leq \rho \|\Delta u_{k-1}(t)\| + (k_p + k_2k_1k_2)k_1 \lambda_2 \|\Delta u_{k+1}(t)\| + k_2k_1k_2 \lambda_2 \|\Delta u_{k+1}(t)\| \]  
(31)

where \(\lambda_2 = \frac{1 - e^{b_2T}}{\lambda - b_2}\) and \(\lambda_2\) will be very small so that can be neglected as \(\lambda\) is big enough. Thus, if the following condition

\[ |I - K_x f(x_{k+1})| \leq \frac{\rho}{1 - k_2k_1k_2 \lambda_2} \leq \rho_1 < 1, \tau \in [0,T] \]

is satisfied, according to the Lemma 1., then yield

\[ \lim_{n \to \infty} \sup \|\Delta u_k(t)\| = 0 \]  
(32)

From (28) we can also get that

\[ \lim_{n \to \infty} \sup \|\Delta u_k(t)\| = 0 \]  
(33)

Therefore, the output errors \(e_k(t)\) will converge to zero as \(k \to \infty\) if the inequality (23) is satisfied.

V. SIMULATION RESULTS

To validate the proposed control strategy, simulations are made based on Peter Corke's robotic vision toolbox[18]. 6-DOF PUMA-560 robot manipulator and central projection camera model are used. The target is defined as nine coplanar points. In the teaching process, the manipulator is actuated by sinusoidal joint velocity and images are recorded as the desired trajectories. Parameters are set as \(K_p = 1, K_x = 0.5, K_d = 0.5\) that satisfies the convergence condition.

For demonstration, the following performance evaluations are used. The norm of error vector at time \(t\) in iteration \(k\) is defined as \(\|e_k(t)\|\) which represents the error in system state space, and the error norm of transform matrix is defined as \(\|T(t)T_k^{-1} - I\|\) which represents the pose error. To show the convergence, \(J_k = \max \left(\|e_k(t)\|\right)\) is used to denote the maximum value of error vector elements.

Fig.5 and Fig.6 show the simulation results. In Fig.5, we choose 4 out of 9 feature points for demonstration, solid lines are the desired trajectories of them and dotted lines are the real trajectories. It can be seen that the real trajectory is closed enough to the desired one after iteration 5. In Fig.6, error convergence in every iteration is shown. From Fig.6(c) we can see that \(J_k\) converges to about \(10^{-2}\) after 20 iterations. Fairly accurate trajectory tracking is achieved.

VI. CONCLUSIONS

In this paper, an open-plus-closed-loop framework is proposed for the vision-based trajectory tracking task of industrial manipulator. System error can be calculated directly from the measurable elements such as homography matrix and feature point coordinates, so that no 3D reconstruction is needed. In the controller design, image Jacobean is not needed, avoiding the singularity problem. System performance can be improved by the iterative learning law which is proven to be convergence. In the end, precise trajectory tracking is achieved after a few iterations. Both simulation and experimental results are given for demonstration.

The proposed method can be easily used in component assembling tasks, and is more flexible to realize compared with conventional teaching-by-doing strategy of industrial manipulators. Future works focus on the repetitive task from arbitrary initial state, which is more productive.
REFERENCES


