

Improved Error Alignment Algorithm for Multi-arm Mechanical Tracking System

Yang Zhang

School of Optoelectronics
Beijing Institute of Technology
Beijing 100081, China
zhangyang6025@gmail.com

Lin Li

School of Optoelectronics
Beijing Institute of Technology
Beijing 100081, China
li_lin@263.net

Abstract- To reduce the optimization parameters in error alignment for the multi-arm mechanical tracking augmented reality (AR) system, the improved error alignment algorithm inspired by Denavit-Hartenberg (D-H) convention commonly used in robotics is proposed. To establish new coordinate systems, the plane geometry theory is introduced. The experimental results show that the improved error alignment algorithm can reduce the optimization parameters, simplify the optimization equations, and shorten the computing time. The errors are optimized effectively using the algorithm.

Keywords- Plane geometry; Error alignment; Augmented reality

I. INTRODUCTION

In robotics applications, Denavit and Hartenberg (D-H) convention is commonly used in hand-eye calibration for selecting frames of reference. The common perpendicular of two lines was the main geometric concept that allowed Denavit and Hartenberg to find a minimal representation [1] [2] [3]. Many research based on D-H convention have been presented in Robotics. Izadbakhsh, A. investigated a dynamic model of the PUMA 560 robot manipulators, based on standard D-H approach [4]. Rahman, M.H. et al. developed a kinematic model of the ExoRob based on modified D-H notations [5]. Maakaroun, S. et al. presented the geometric model of a new narrow tilting car based on the modified D-H geometric description [6]. Fueanggan, S. designed the dynamics and kinematics simulation software for education in which the kinematics model to consider Forward kinematics based on D-H Transformation [7]. Parasuraman, S. used D-H method for the positional analysis to determine the end position in 3D space [8]. In this paper, an improved error alignment algorithm inspired by D-H convention for a multi-arm mechanical tracking augmented reality (AR) system is proposed.

The paper is organized in 5 sections. Section II presents the previous error alignment method of the multi-arm augmented reality (AR) system. The system structure and tracking algorithm are introduced in Section II. A. The previous error alignment method is described in Section II. B. Section III presents the improved algorithm based on “plane theory” and describes coordinate systems transformation. Section IV presents the experimental results.

II. PREVIOUS WORK AND ERROR ALIGNMENT METHOD

A. The System Architecture and Tracking Algorithm

In order to achieve high precision registration in relatively small range, we have developed a novel mechanical tracking tabletop augmented reality (AR) system that photoelectric encoders read joints’ rotation angles (shown in Fig. 1).

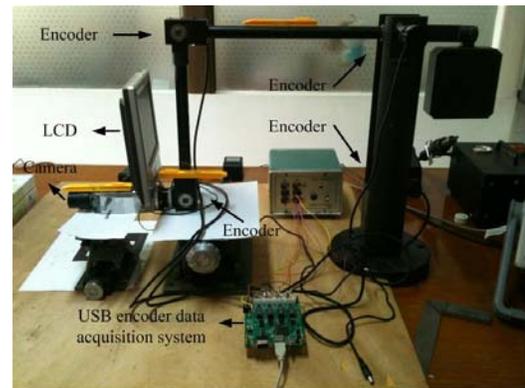


Fig. 1 System architecture

To obtain the coordinate systems transformation from the bottom of the mechanical system to the camera coordinate system, coordinate systems are established on every joint (shown in Fig. 2), and are represented in (1). M_{12} , M_{23} , M_{34} , M_{45} , M_{5c} are the rotation and translation matrixes between two adjacent joints.

$$\Delta M = M_{5c} \cdot M_{45} \cdot M_{34} \cdot M_{23} \cdot M_{12} \quad (1)$$

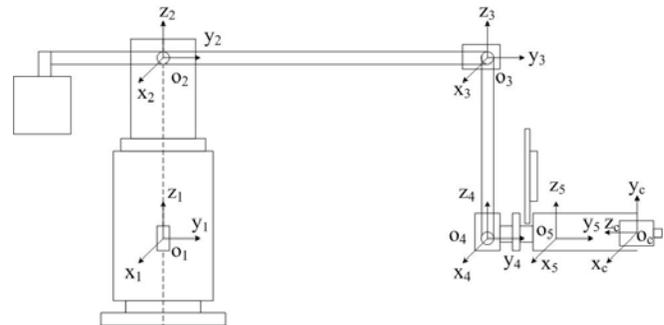


Fig. 2 The transformation of every joint coordinate system

B. Previous Error Alignment Method

The coordinate systems transformation between the camera and the bottom of the mechanical system should include the extra rotation matrixes formed by twist angles between the adjacent joints and the extra translation matrixes formed by arm measurement errors. The twist angles and the length measurement errors are seemed as the optimization parameters.

The extra rotation of every joint is in three directions and three matrixes should be added in the transformation of the adjacent coordinate systems as (2).

$$\begin{aligned} \Delta M = & (M_1 \cdot M_2 \cdot M_3) \cdot M_{5c} \cdot (M_4 \cdot M_5 \cdot M_6) \cdot M_{45} \\ & \cdot (M_7 \cdot M_8 \cdot M_9) \cdot M_{34} \cdot (M_{10} \cdot M_{11} \cdot M_{12}) \\ & \cdot M_{23} \cdot (M_{13} \cdot M_{14} \cdot M_{15}) \cdot M_{12} \end{aligned} \quad (2)$$

“fmincon” function of matlab toolbox is used as the main optimization tool. The principle is nonlinear least square minimization. There are 22 parameters to be optimized result in that the calculating is time-consuming and the optimization results are even not convergent.

III. IMPROVED ERROR ALIGNMENT METHOD

A. The “plane theory” for simplifying the Optimization

Using the previous optimization method, it is difficult to make the results convergent and is time-consuming. In this paper, new method is explored to make the results convergent and cost lower time. Using plane geometry theory, the new coordinate systems are established. The optimization parameters are reduced based on coordinate systems set up in this way. The “plane theory” is proposed based on three assumptions as follows.

1) Every rotary encoder has an axis. The rotation of the encoder is in the plane which is vertical to the axis (shown in Fig. 3).

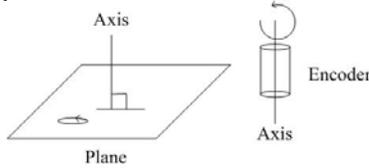


Fig. 3 Encoder rotate along the axis and in the plane

2) When two planes intersect, the line of intersection is on both of the two planes. This intersecting line is used as one axis to set up the transition coordinate system (shown in Fig. 4).

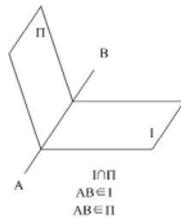


Fig. 4 Two planes intersect and the line of intersection belongs to both planes

3) Two axes of the coordinate system are set up on the plane. The other axis of the coordinate system is along the direction of the axis of encoder, which is vertical to the plane (shown in Fig. 5).

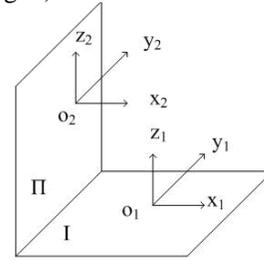


Fig. 5 One axis established along the encoder axis and the other two axes on the plane vertical to the encoder axis according to right hand coordinate system

B. The Transition from Plane 1 to Plane 2

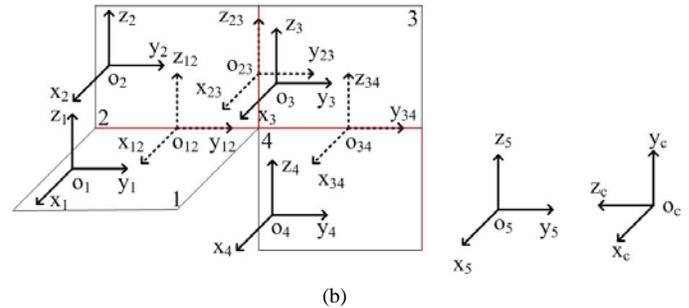
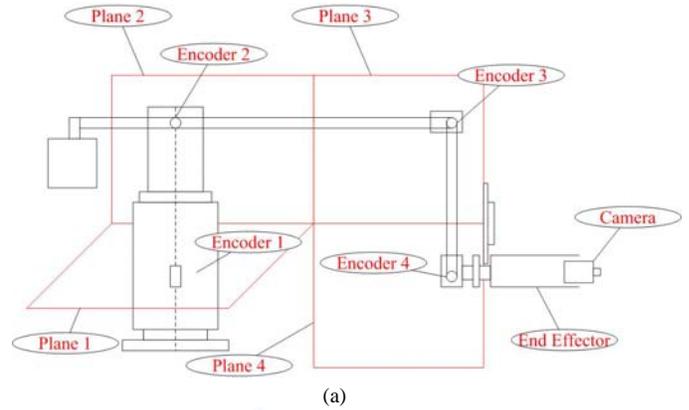


Fig. 6 (a) The planes vertical to the encoders' axes (b) The planes intersect and the coordinate systems established on the planes

About the establishment of the coordinate systems before, twist of adjacent links are in x , y and z directions, so to each link, three coordinate systems should be added. Each coordinate system with a rotation angle parameter to be optimized lead to optimization results are hard to converge. In order to make the optimization be more rational, planes geometry theory is used to establish new coordinate systems.

As Fig. 6 shows, plane 1 are vertical to the axis of encoder 1 and plane 2 are vertical to the axis of encoder 2. In ideal situation, the axis of encoder 1 is vertical to the axis

of encoder 2. In real situation, the separation angle is not 90 degree (shown in Fig. 7), so that there is an extra rotation matrix in the coordinate systems transformation (shown in Fig. 8).

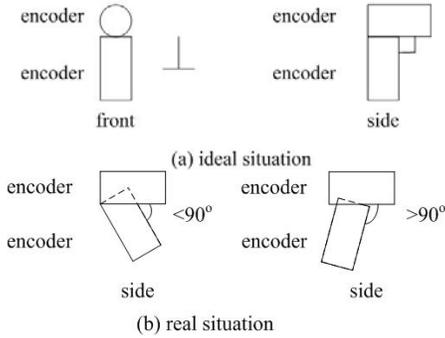


Fig. 7 Two encoders in ideal situation and in real situation

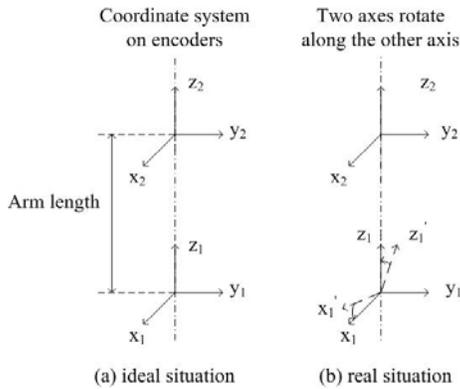


Fig. 8 The coordinate systems in ideal situation and in real situation

Plane 1 and plane 2 intersect. The co-owned y axis (y_{12}) is defined along the line of intersection. z_1 is set up along the direction of the axis of the encoder 1. x_2 is along the axis of encoder 2. The other axes of the coordinate systems (x_1 and z_2) are set up according to right-hand orthogonal coordinate systems. Because axis 1 and axis 2 are not vertical in real situation, coordinate system 1 rotates anticlockwise for α angle round y axis to transform to coordinate system 2 (shown in Fig. 6).

The auto-rotation of encoder 1 is added. Coordinate system 1 rotates round z_1 for $\angle 1$ angle and translates along the first arm is added. The arm length is 310mm. An error length l_1 is added. The whole transformation matrix from coordinate system 1 to 2 is as (3).

$$M_{12} = \begin{bmatrix} \cos \angle 1 & \sin \angle 1 & 0 & 0 \\ -\sin \angle 1 & \cos \angle 1 & 0 & 0 \\ 0 & 0 & 1 & -310+l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

C. The Transition from Plane 2 to Plane 3

As Fig. 6 shows, in ideal situation, plane 2 is parallel with plane 3. In real situation, there is a separation angle between the axes of encoder 2 and encoder 3. Plane 2 and plane 3 intersect in this situation. Coordinate system 2 rotates anticlockwise for β angle round x axis to transform to coordinates 12, which is set up on the line of intersection, then rotates anticlockwise for γ angle round z axis to coordinate system 3.

The auto-rotation of encoder 2 is added. Coordinates 2 rotates round x_2 for $\angle 2$ angle and translates along the second arm is added. The arm length is 400mm. An error length l_2 is added. The whole transformation matrix from coordinates 1 to 2 is as (4).

$$M_{23} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \angle 2 & \sin \angle 2 & -400+l_2 \\ 0 & -\sin \angle 2 & \cos \angle 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \beta & \sin \beta & 0 \\ 0 & -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \gamma & \sin \gamma & 0 & 0 \\ -\sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

D. The Transition from Plane 3 to Plane 4

It is like the transition from plane 2 to plane 3. As Fig. 6 shows, in ideal situation, plane 3 is parallel with plane 4. In real situation, there is a separation angle between the axes of encoder 3 and encoder 4. Plane 3 and plane 4 intersect in this situation. Coordinates 3 rotates anticlockwise for θ_1 angle round x axis to transform to coordinates 34, which is set up on the line of intersection, then rotates anticlockwise for θ angle round y axis to coordinates 4.

The rotation of encoder 3 is added. Coordinates 3 rotates round x_3 for $\angle 3$ angle and translates along the third arm is added. The arm length is 294mm. An error length l_2 is added. The whole transformation matrix from coordinates 3 to 4 is as (5).

$$M_{34} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \angle 3 & \sin \angle 3 & 0 \\ 0 & -\sin \angle 3 & \cos \angle 3 & 294 + l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_1 & \sin \theta_1 & 0 \\ 0 & -\sin \theta_1 & \cos \theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

E. The Transformation from Coordinate system 4 to camera

The camera coordinate system is set up according to OpenGL (shown in Fig. 9). y axis is along the up direction. z axis is opposite to the direction of the line of sight.

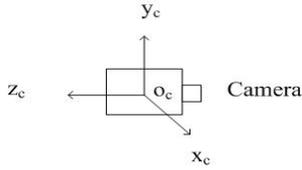


Fig. 9 The camera coordinate system

The position of the origin and three directions of camera coordinate system are unknown. Coordinates 4 rotates round three directions to transform to camera coordinate system. As Fig. 6 shows, firstly, according to rough estimation, Coordinates 4 rotates anticlockwise for 90 degree angle round x axis to transform to general directions of the camera coordinate system, and then rotates anticlockwise for δ_1 angle round x axis, and then rotates anticlockwise for δ_2 angle round y axis, and then rotates anticlockwise for δ_3 angle round z axis to transform to the correct directions of the camera coordinate system.

The auto-rotation of encoder 4 is added. Coordinates 4 rotates round x_4 for $\angle 4$ angle and translates along the forth arm is added. The arm length is 221mm. An error length l_5 is added. Because there is a translation from the center of the camera and the center of the encoder, there are errors need to be aligned in three directions, x , y and z . So the other error length l_4 and l_6 are added. The whole transformation matrix from coordinates 4 to camera is as (6).

$$M_{4c} = \begin{bmatrix} 1 & 0 & 0 & l_4 \\ 0 & \cos \angle 4 & \sin \angle 4 & l_5 - 221 \\ 0 & -\sin \angle 4 & \cos \angle 4 & l_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \delta_1 & \sin \delta_1 & 0 \\ 0 & -\sin \delta_1 & \cos \delta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \delta_2 & 0 & -\sin \delta_2 & 0 \\ 0 & 1 & 0 & 0 \\ \sin \delta_2 & 0 & \cos \delta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$\begin{bmatrix} \cos \delta_3 & \sin \delta_3 & 0 & 0 \\ -\sin \delta_3 & \cos \delta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The whole transformation from the camera coordinate system to the bottom of the mechanical tracking system coordinate is as (7).

$$\Delta M = M_{4c} \cdot M_{34} \cdot M_{23} \cdot M_{12} \quad (7)$$

IV. EXPERIMENT RESULTS

The contrast between the previous optimization theory and the “plan theory” shows that the optimization parameters and the matrixes are reduced obviously using “plane theory” (show in Table I). The optimization result is convergent through the improved algorithm. Using the optimized parameters, the accurate registration in AR is achieved.

TABLE I Contrast of different methods

	previous method	“plane theory”
Number of optimization parameters	22	14
Number of matrixes	20	13

V. CONCLUSION

In this paper, the plane geometry theory is used to solve the error alignment problem for the mechanical tracking AR system. In the establishment of the coordinate systems using previous method, on each joint, 4 coordinate systems should be established, which include three rotations due to error caused by twisting of adjacent links in three directions, and natural rotation of the links because of rotation of rotary encoders. In the establishment of the coordinate systems, using improved method, on each joint only one coordinate system should be established that is the natural rotation of the links because of rotation of rotary encoders. Two axes of a coordinate system compose a plane. Two adjacent planes intersect. One coordinate system is established on the intersection line. The establishment of coordinate systems is greatly reduced.

The improved error alignment algorithm can reduce the optimization parameters, simplify the optimization equation,

and shorten the computing time. This algorithm can be used in robot coordinate systems calibration.

REFERENCES

- [1] J. Denavit, R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices", *Trans ASME J. Appl. Mech*, 23, pp. 215-221, 1955.
- [2] R. S. Hartenberg, J Denavit, "Kinematic synthesis of linkages McGraw-Hill", New York, NY, 1964.
- [3] http://en.wikipedia.org/wiki/Denavit-Hartenberg_Parameters
- [4] Izadbakhsh, A., "Closed-form dynamic model of PUMA 560 robot arm", *4th International Conference on Autonomous Robots and Agents*, Wellington, 2009, pp. 675-680, 10-12 Feb. 2009.
- [5] Rahman, M.H. ; Saad, M. ; Kenne ; , J.P. ; Archambault, P.S. , "Exoskeleton robot for rehabilitation of elbow and forearm movements", *2010 18th Mediterranean Conference on Control & Automation (MED)*, pp. 1567-1572, 23-25 June 2010.
- [6] Maakaroun, S. ; Khalil, W. ; Gautier, M. ; Chevrel, P. ; "Geometric model of a narrow tilting CAR using robotics formalism", *2010 15th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 377-382, Miedzyzdroje, 23-26 Aug. 2010.
- [7] Fueanggan, S., Chokchaitam, S., "Dynamics and Kinematics Simulation for Robots", *International Association of Computer Science and Information Technology – Spring Conference*, 2009, pp. 136-140, Singapore, 17-20 April 2009.
- [8] Parasuraman, S. ; Oyong, A.W. ; Ganapathy, V. , "Development of robot assisted stroke rehabilitation system of human upper limb", *IEEE International Conference on Automation Science and Engineering*. pp. 256-261, Bangalore, 2009