EN3563 Robotics
Laboratory Experiment 02

Title: Homogeneous Transformations using Robotics Toolbox

1. Introduction

The relationship between two rigid bodies can be established by first attaching a coordinate frame to each rigid body and describing the relative position (translation) and relative orientation (rotation) between the two coordinate frames (Fig. 1). We combine rotation and translation to define rigid motion given by a homogeneous transformation matrix.

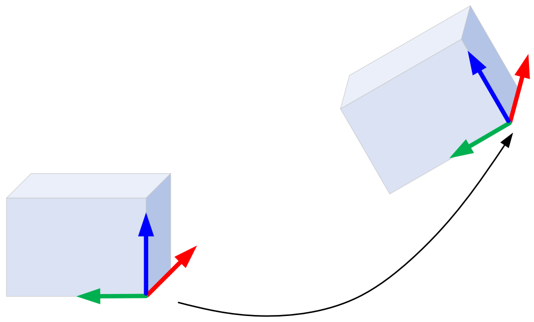


Figure 1: Establishing a relationship between two rigid bodies

The Robotics Toolbox is a MATLAB toolbox software that supports research and teaching into arm-type and mobile robotics. It contains functions and classes to represent orientation and pose in 2D and 3D (SO(2), SE(2), SO(3), SE(3)) as matrices, quaternions, twists, triple angles, and matrix exponentials. You are expected to have correctly configured the Robotics Toolbox to proceed with the remainder of this laboratory experiment.

From this experiment you will learn the following:

- Usage of MATLAB Robotics Toolbox
- Reinforce the understanding of homogeneous transformation.

2. Theory

This section describes the underlying theories associated with rigid motion and homogeneous transformations.

2.1. Homogeneous Coordinates

Homogeneous coordinates are a way of representing N-dimensional coordinates with N+1 numbers. Equation (1) converts a 3D point into its homogeneous coordinates by adding a fourth element '1'. Homogeneous coordinates provide a standard to perform certain standard operations on points in Euclidean space using matrix multiplication.

$$p^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{3\times1} \quad \rightarrow \quad P^0 = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{4\times1} \tag{1}$$

2.2. Homogeneous Transformation Matrix

A homogeneous transformation matrix (eq. (2)) is a 4×4 matrix that belongs to the special Euclidean group $SE(3)$ to represent rigid motion.

$$H = \begin{bmatrix} R_{3\times3} & t_{3\times1} \\ 0_{1\times3} & 1_{1\times1} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$R \in SO(3)$ is a rotation matrix, and $t \in R^3$ is a translational vector.

2.3. Homogeneous Transformation

Using a homogeneous transformation matrix H_1^0 and homogeneous coordinates, a point p can be transformed from one frame to another.

$$P^0 = H_1^0 P^1 \tag{3}$$

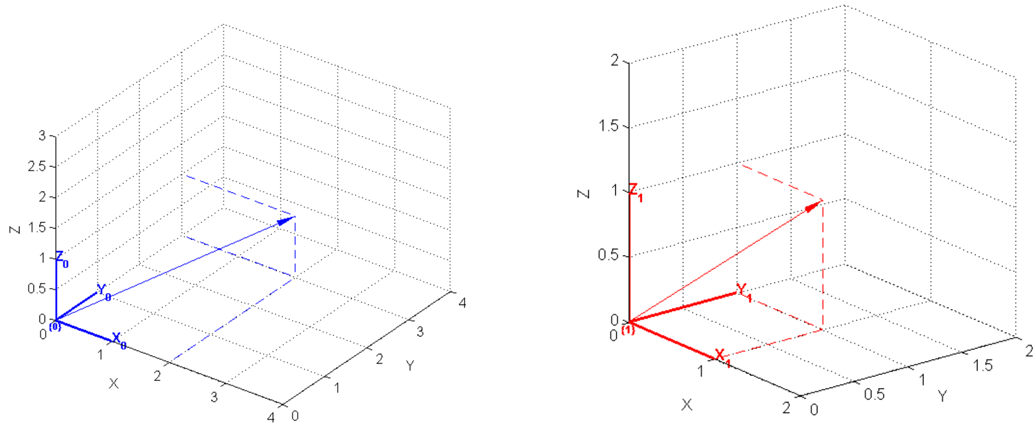
2.4. Inverse Homogeneous Transformation Matrix

The inverse of the homogeneous transformation matrix is given by eq. (4).

$$H^{-1} = \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix} \tag{4}$$

3. Procedure

Figure 2(a) shows a position vector q^0 using a blue color arrow with respect to the coordinate frame $\{0\}$, and Figure 2(b) shows another position vector p^1 using a red color arrow with respect to the coordinate frame $\{1\}$. The origin of frame $\{1\}$ is positioned at the tip of the blue color q^0 position vector. The orientation of frame $\{1\}$ can be obtained by rotating the default coordinate frame $\{0\}$, 90^0 around its z axis.



(a) q^0 in $\{0\}$ frame *(b) p^1 in $\{1\}$ frame*

Figure 2: different position vectors

Follow the subsequent procedure using notation as described in section 5, where necessary.

3.1. Visualize coordinate system $\{0\}$ on a new MATLAB figure preserving prescribed colors. Limit the plot area to X: $[0, 4]$, Y:$[0, 4]$, and Z:$[0,3]$. Enable the grid.
3.2. Obtain the rotation matrix R_1^0 and translation vector t_1^0 that would characterize the rigid motion from coordinate frame $\{0\}$ to coordinate frame $\{1\}$.
3.3. Visualize q^0 in the figure using blue color.
3.4. Obtain the homogeneous transformation matrix H_1^0 using Robotics Toolbox for the rigid motion described in 3.2. Visualize the coordinate frame $\{1\}$ using red color on the same figure.
3.5. Find p^0 and visualize it in the same figure using green color.
3.6. Visualize p^1 in the same figure using red color.
3.7. Use a *new* MATLAB figure to visualize the coordinate system $\{1\}$ and p^1 using red color. Limit the plot area to X: $[-4, 2]$, Y:$[-1, 3]$, and Z:$[-2,2]$. Enable the grid.
3.8. Now that you are in frame $\{1\}$, you are required to visualize coordinate frame $\{0\}$ on this new figure. Obtain the homogeneous transformation matrix H_0^1.
3.9. Visualize frame $\{0\}$ with blue color.
3.10. Find t_0^1 and visualize it on the figure with blue color.
3.11. Visualize a green arrow from the tip of p^1 to the origin of frame $\{0\}$.
3.12. You are given a MATLAB script* to generate the 3D environment of a table (height 1m, width 1m and length 1m), a box (every side 20cm) and a camera (2m directly above the table). You are required to plot the coordinate frames $\{0\},\{1\},\{2\},\{3\}$ in your figure and fill the table in 8 of the answer sheet. Your completed MATLAB figure should look exactly like Figure 3.

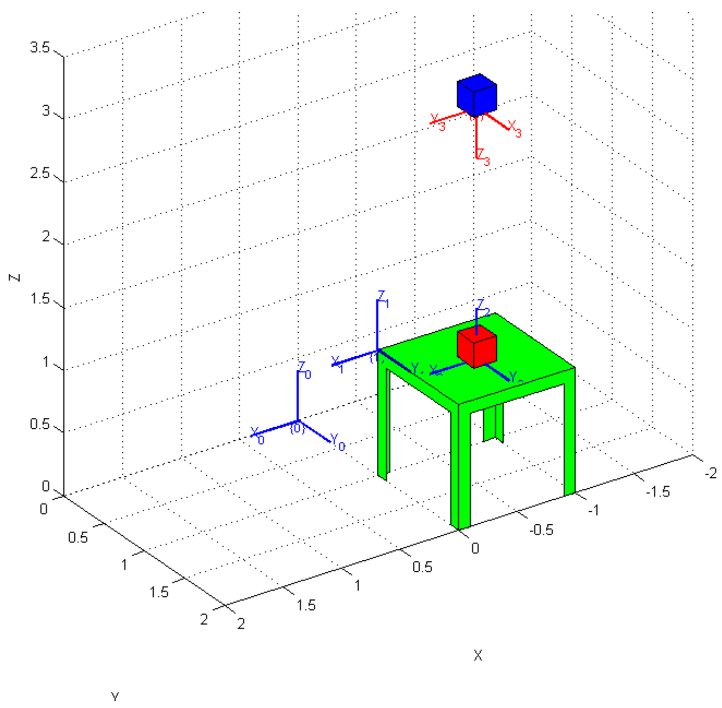


Figure 3: 3D environment for visual servoying

*Please use the following script to generate the 3D environment.

```
close all;
clear all;

figure;
xlabel('x'); ylabel('y'); zlabel('z');

trplot(eye(4),'color','b','frame','0','length',0.4,'thick',2);
hold on; grid on; axis equal;
axis([-2 2 0 2 0 3.5]);

fill3([0 0 -1 -1],[1 2 2 1],[1 1 1 1],'g');
fill3([0 0 0 0 0 0 0 0],[1 1.1 1.1 1.9 1.9 2 2 1],[0 0 0.9 0.9 0 0 1 1],'g');
fill3([-1 -1 -1 -1 -1 -1 -1 -1],[1 1.1 1.1 1.9 1.9 2 2 1],[0 0 0.9 0.9 0 0 1 1],'g');
fill3([0 -0.1 -0.1 -0.9 -0.9 -1 -1 0],[2 2 2 2 2 2 2 2],[0 0 0.9 0.9 0 0 1 1],'g');
fill3([0 -0.1 -0.1 -0.9 -0.9 -1 -1 0],[1 1 1 1 1 1 1 1],[0 0 0.9 0.9 0 0 1 1],'g');

fill3([-0.4 -0.4 -0.6 -0.6],[1.4 1.6 1.6 1.4],[1 1 1 1],'r');
fill3([-0.4 -0.4 -0.6 -0.6],[1.4 1.6 1.6 1.4],[1.2 1.2 1.2 1.2],'r');
fill3([-0.4 -0.6 -0.6 -0.4],[1.6 1.6 1.6 1.6],[1 1 1.2 1.2],'r');
fill3([-0.4 -0.6 -0.6 -0.4],[1.4 1.4 1.4 1.4],[1 1 1.2 1.2],'r');
fill3([-0.4 -0.4 -0.4 -0.4],[1.4 1.6 1.6 1.4],[1 1 1.2 1.2],'r');
fill3([-0.6 -0.6 -0.6 -0.6],[1.4 1.6 1.6 1.4],[1 1 1.2 1.2],'r');

fill3([-0.4 -0.4 -0.6 -0.6],[1.4 1.6 1.6 1.4],[3 3 3 3],'b');
fill3([-0.4 -0.4 -0.6 -0.6],[1.4 1.6 1.6 1.4],[3.2 3.2 3.2 3.2],'b');
fill3([-0.4 -0.6 -0.6 -0.4],[1.6 1.6 1.6 1.6],[3 3 3.2 3.2],'b');
fill3([-0.4 -0.6 -0.6 -0.4],[1.4 1.4 1.4 1.4],[3 3 3.2 3.2],'b');
fill3([-0.4 -0.4 -0.4 -0.4],[1.4 1.6 1.6 1.4],[3 3 3.2 3.2],'b');
fill3([-0.6 -0.6 -0.6 -0.6],[1.4 1.6 1.6 1.4],[3 3 3.2 3.2],'b');
```

4. MATLAB Robotics Toolbox Reference

4.1. Position Vector

`plot_arrow`

Draw an arrow in 2D or 3D

4.2. Rotation Matrix

`rot2`: SO(2) rotation matrix
`rot3`: SO(3) rotation matrix

4.3. Coordinate Frame

`trplot2`: 2D Coordinate frame
`trplot`: 3D Coordinate frame

<u>Related commands</u>
`hold on`: multiple frames can be added
`axis`: limit the plot area
`grid`: grid can be added to the plot

4.4. Rotations

`rotx`: SO(3) rotation about X axis
`roty`: SO(3) rotation about Y axis
`rotz`: SO(3) rotation about Z axis

4.5. Conversions

`e2h`: Euclidian to homogeneous
`h2e`: homogeneous to Euclidian
`rt2tr`: convert rotation and translation to homogeneous transform
`tr2rt`: convert homogeneous transform to rotation and translation
`r2t`: convert rotation matrix to homogeneous transform

5. Notation

Use a meaningful notation in MATLAB similar to the following:

Notation	Meaning
p_in_a	Point **p** represented in frame {a}
R_a_in_b	Orientation of frame {a} with respect to frame {b}
R_x_30	30^0 rotation about X axis

Answer Sheet

Index No:

1. Homogeneous transformation matrix H_1^0 for 3.4.

2. MATLAB code for 3.1 ~ 3.6.

3. Final output MATLAB figure for the operations in 3.1 ~ 3.6.

4. Homogeneous transformation matrix H_0^1 for 3.8.

5. t_0^1 for 3.10.

6. MATLAB code for 3.7 ~ 3.11.

7. Final output MATLAB figure for the operations in 3.7 ~ 3.11.

8. Homogeneous transformation table.

Requirement	MATLAB script to satisfy the requirement	Homogeneous transformation matrix result
$o_0x_0y_0z_0$ to $o_1x_1y_1z_1$		
$o_0x_0y_0z_0$ to $o_2x_2y_2z_2$		
$o_0x_0y_0z_0$ to $o_3x_3y_3z_3$		