```
function [trajectory,csv_list] = TrajectoryGenerator(Tse_initial,Tsc_initial,Tsc_final,Tce_grasp,Tce_standoff,k)
%TRAJECTORYGENERATOR    Reference trajectory generator for a pick-and-place
%                       task with the KUKA youBot.
%
%   [trajectory,csv_list] =
%   TRAJECTORYGENERATOR(Tse_initial,Tsc_initial,Tsc_final,Tce_grasp,Tce_standoff,k)
%   produces a cell array (trajectory) containing SE(3) matrices
%   representing end-effector configuration matrices and a matrix
%   (csv_list) written to a csv file to be visualized in CoppeliaSim Scene
%   8 associated with the book "Modern Robotics: Mechanics, Planning, and
%   Control," Kevin Lynch and Frank Park, Cambridge University Press, 2017.
%
%   Input           Value
%   Tse_initial     The initial SE(3) configuration of the end-effector in
%                   the space frame
%   Tsc_initial     The inital SE(3) configuration of a block in the space
%                   frame.
%   Tsc_final       The desired SE(3) configuration of the block in the
%                   space frame.
%   Tce_grasp       The desired SE(3) configuration matrix of the
%                   end-effector in the block frame while grasping.
%   Tce_standoff    The stand-off/stand-by SE(3) configuration of the
%                   end-effector in the block frame before grasping.
%   k               Number of reference configurations per time step.
%
%   Output          Value
%   trajectory      An N-long cell array containing
%                   the SE(3) configuration of the end-effector in the
%                   space frame for each time step.
%   csv_list        A N-by-13 matrix of configuration variables for the
%                   KUKA youBot written to a csv file.
%
%   See also TRAJECTORYTEST, WRAPPER.

%   Written by David Lim for the MAE 204 Final Project WI25. Last modififed
%   on 03/08/25.

dt = 0.01;  % time step
method = 5; % time scaling method for trajectory
T1 = 5;  % duration of intitial to standoff
T2 = 2;  % duration of standoff to grasp
T3 = 1;  % duration of grasp
T4 = T2;  % duration of grasp to standoff
T5 = T1;  % duration of standoff grasp to standoff ungrasp
T6 = T2;  % duration of standoff to ungrasp
T7 = T3;  % duration of ungrasp
T8 = T2;  % duration of ungrasp to standoff

T_list = [T1 T2 T3 T4 T5 T6 T7 T8]';  % list of segment durations in seconds
N_list = T_list*k/dt;  % list of segment durations in steps
N_sum = cumsum(N_list);  % list of time stamps in steps
N = N_sum(end);  % total number of steps
gripper_state = zeros(N,1);  % list of gripper states
gripper_state(N_sum(2):N_sum(6)) = 1;  % assign closed states

% list of SE(3) configuration waypoints in space frame
Tse_list = {Tse_initial ...
            Tsc_initial*Tce_standoff ...
            Tsc_initial*Tce_grasp ...
            Tsc_initial*Tce_grasp ...
            Tsc_initial*Tce_standoff ...
            Tsc_final*Tce_standoff ...
            Tsc_final*Tce_grasp ...
```

```
            Tsc_final*Tce_grasp ...
            Tsc_final*Tce_standoff};

% generate reference trajectory list
trajectory = cell(1,N);
trajectory(1:N_list(1)) = ScrewTrajectory(Tse_list{1},Tse_list{2},T_list(1),N_list(1),method);
for j = 2:8
    trajectory(N_sum(j-1)+1:N_sum(j)) = ScrewTrajectory(Tse_list{j},Tse_list{j+1},T_list(j),N_list(j),method);
end

% generate csv file
csv_list = zeros(N,13);
for n = 1:N
    [R,p] = TransToRp(trajectory{n});
    R = R';
    csv_list(n,1:12) = [R(:)' p'];
end
csv_list(:,13) = gripper_state;
writematrix(csv_list,'trajectory.csv')
end
```

Published with MATLAB® R2024a