Inverse Kinematics Scripts

Purpose

This set of scripts provides inverse kinematics solutions tailored for the UR10 robot arm, complemented by a custom actuator. It's designed to facilitate the visualization of the robot grasping and positioning boxes conveyed by a belt system.

Prerequisites

Unity Setup:

Ensure your Unity project is initialized using the 3D URP (Universal Render Pipeline) template. This script was developed and tested for Unity Editor version `2021.3.19f1` and as such, stability of this script in future versions is not guaranteed. The scripts can be downloaded, along with a demo project, can be downloaded from the DTLab GitLab page and then copied to the new Unity project. The essential scripts for Inverse Kinematics, without the pick-and-place functionality, are:

- `IKController` -> to control the robot arm's movement
- `DisableGravityForJoints` -> to override the physics built into the URDF model
- `MoveAlongPath` -> to move the box along the conveyor belt

Required Packages:

For effective robot arm simulation and interaction, integrate the following packages:

- ROS TCP Connector
- Unity Robotics Visualizations
- URDF Importer

All the above packages can be sourced and installed from Unity Robotics Hub on GitHub. Alternatively, they can be imported from locally cloned repositories housing the packages.

Robot Arm Integration:

- Using a URDF File: If your robot arm configuration is stored in a URDF file, navigate to the asset within Unity. Right-click and choose "Import Robot from Selected URDF file". This action will generate a 3D visual of the robot arm, incorporating the designated joints and their inherent physical constraints.
- Without a URDF File: If you possess only a 3D render of the robot, devoid of joint articulation, consider importing this model into Blender. Within Blender, introduce "bones" to your model. These bones will enable the various robot segments to exhibit mobility and flexibility.

Setup and Installation

1. **Script Integration:** Begin by adding the IK script to your Unity project.
2. **Setting up the IK Controller:**
   - Create a new empty GameObject within your scene.
   - Rename this GameObject to IK Controller.
   - Attach the IK Controller script to this GameObject.
3. **Configuration:**
   - Assign the appropriate GameObjects to the Actuator and IK Target fields.
   - For scenarios requiring multiple targets, ensure you populate the Target Array with the relevant GameObjects in the order they should be referenced.
4. **Joints Array Setup:**
   - Populate the Joints array with all individual joints from the robot arm.
   - Start with the base link (root of the arm) as Element 0 in the array.
   - Conclude the array with the actuator as the final element.

By following these steps, you will have successfully set up the IK Controller to interact with your robot arm in Unity.

Revision #4
Created 4 September 2023 12:54:19 by Matilda Fogato
Updated 26 February 2024 10:58:09 by Matilda Fogato