KUKA SUNRISE TOOLBOX (KST)

CONTROL IIWA • FROM MATLAB

Open source:
KST is an open source project. It is a MATLAB toolbox for controlling KUKA iiwa robots from external PC. The toolbox is available in github, and is provided under MIT license:
https://github.com/Modi1987/KST-Kuka-Sunrise-Toolbox

Easy to use:
Control iiwa using familiar scripting language of MATLAB.

Includes various tutorial examples:
Includes scripts for controlling iiwa using 3d space mouse, joystick, cad files and more. See next page.

Easy integration of external sensors:
Easy to integrate sensors and external controls.

Supported with video tutorials on YouTube:
Freely accessible, links are available in the repo webpage.

Sponsored by:

UNIVERSIDADE D COIMBRA

Email: ms@uc.pt

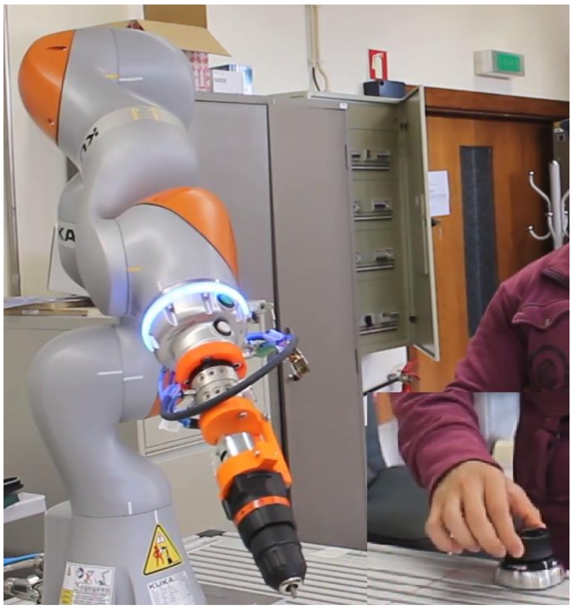
Example: objects recognition and grasping from MATLAB. KUKA iiwa controlled using KST.

Freely available under MIT license at github, visit the repo webpage from your smartphone.

URL in QR code

KUKA SUNRISE TOOLBOX (KST)

CONTROL IIWA • FROM MATLAB

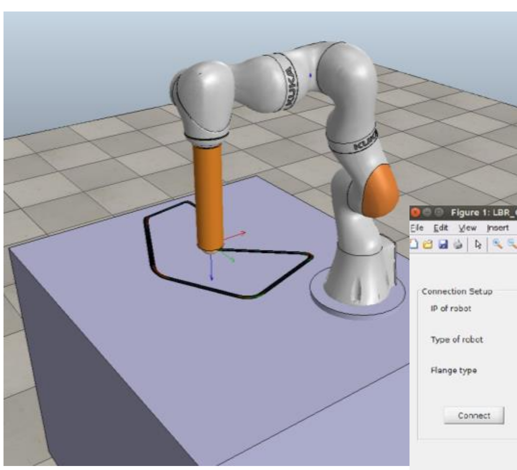


Included example: control iiwa on-the-fly using 3D space mouse

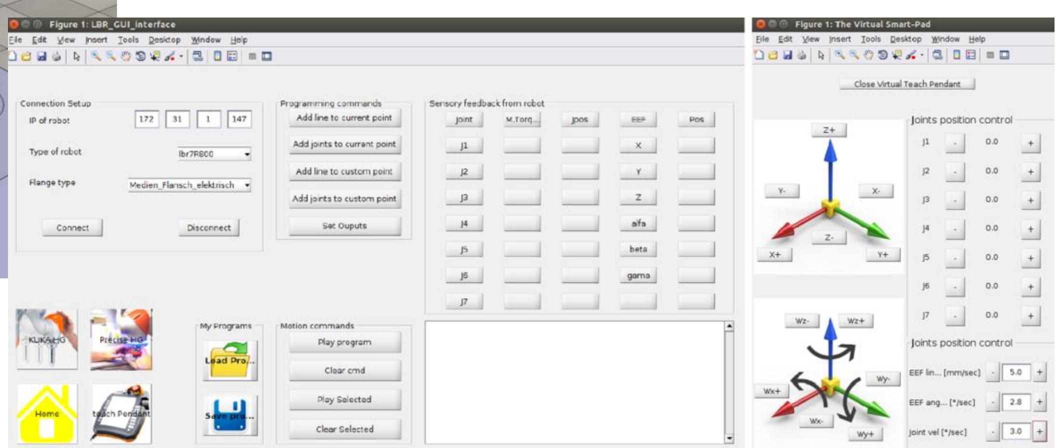
Algorithm 1: Drawing a rectangle (MATLAB code)

```
% Connect to a robot with IP 172.31.1.147
[tKuka,flag]=connectToKuka('172.31.1.147')
% Define the robot initial configuration in joint space
pos={0, -pi/180*10, 0, -pi/180*100, pi/180*90, pi/180*90, 0}
% Define the override velocity of the joints motion
relVel=0.2
% Move the robot to the specified initial configuration
movePTPJointSpace(tKuka, pos, relVel)
% Define the relative position of the first rectangle
point (100 mm away in Z direction)
distPos={0,0,100}
% Define end-effector linear velocitY (mm/sec)
vel=30
% The end-effector is moved down towards the upper
side of the box in relative motion
movePTPLineEefRelEef(tKuka, distPos, vel)
% The length of the sides of the rectangle are (w1,w2)
w1=100; w2=125;
% Move the robot considering the relative position of
each node of the rectangle relative to the frame of
the end-effector
distPos={w1,0,0}
movePTPLineEefRelEef(tKuka , distPos, vel)
distPos={0,w2,0}
movePTPLineEefRelEef(tKuka , distPos, vel)
distPos={-w1,0,0}
movePTPLineEefRelEef(tKuka , distPos, vel)
distPos={0,-w2,0}
movePTPLineEefRelEef(tKuka , distPos, vel)
```

Control iiwa using the familiar scripting language of MATLAB



Included example: Integrate 3D simulators while controlling the real robot on-the-fly.



Included example: integrate GUI interfaces for controlling iiwa