

Deep learning-based Machine Vision for the Task of Grasping Chemical Hardware

Машинное зрение на основе глубокого обучения для захвата химического оборудования

Coursework

Presented by Pongsapas Watana

Advisor: Okunev Alexey G.

22-Dec-2020

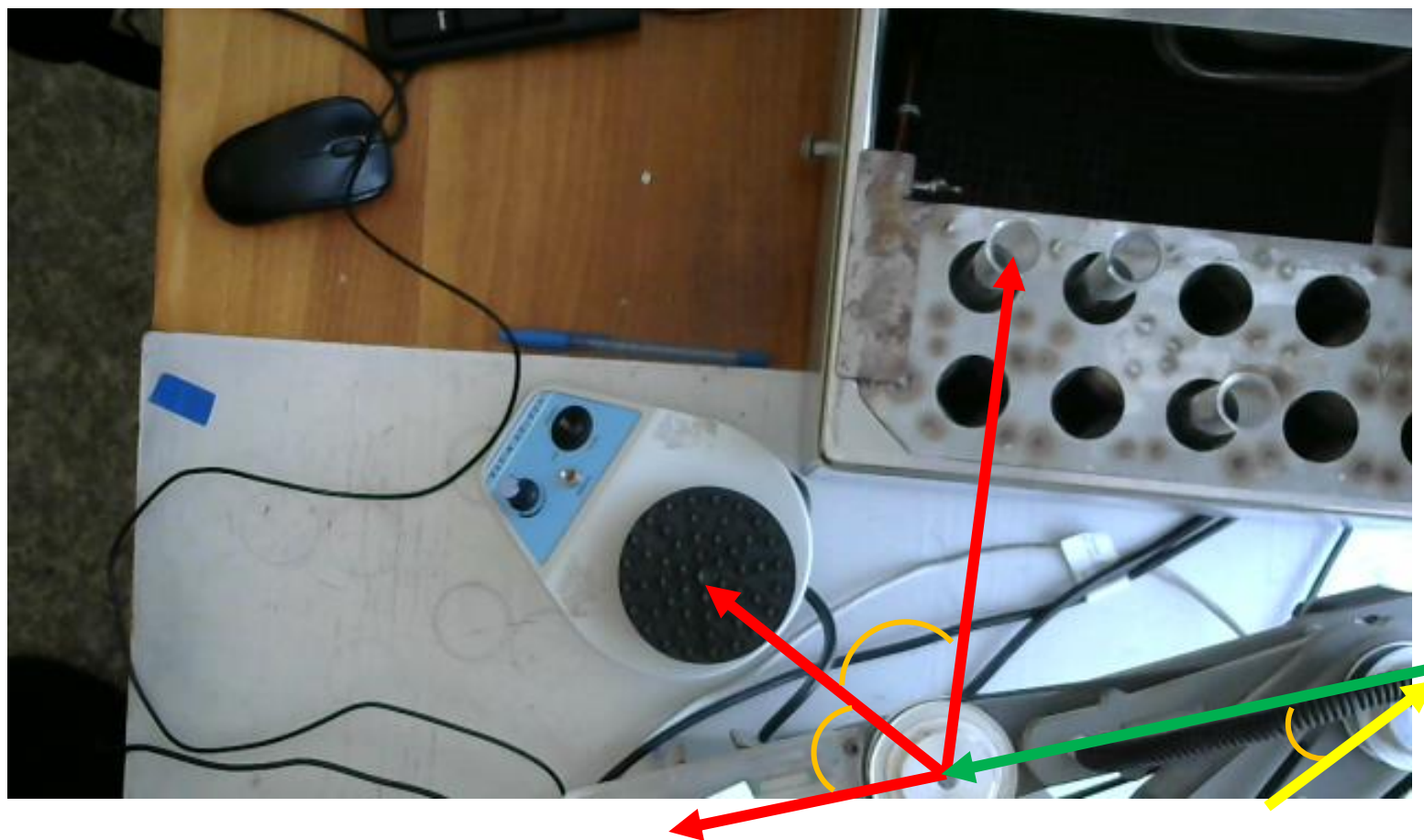
Current Progress

- Object annotation was done
- Study possibility in coding the object detection with MMDetection on GoogleColab
- Assess and select a robot simulator for implementing the algorithm

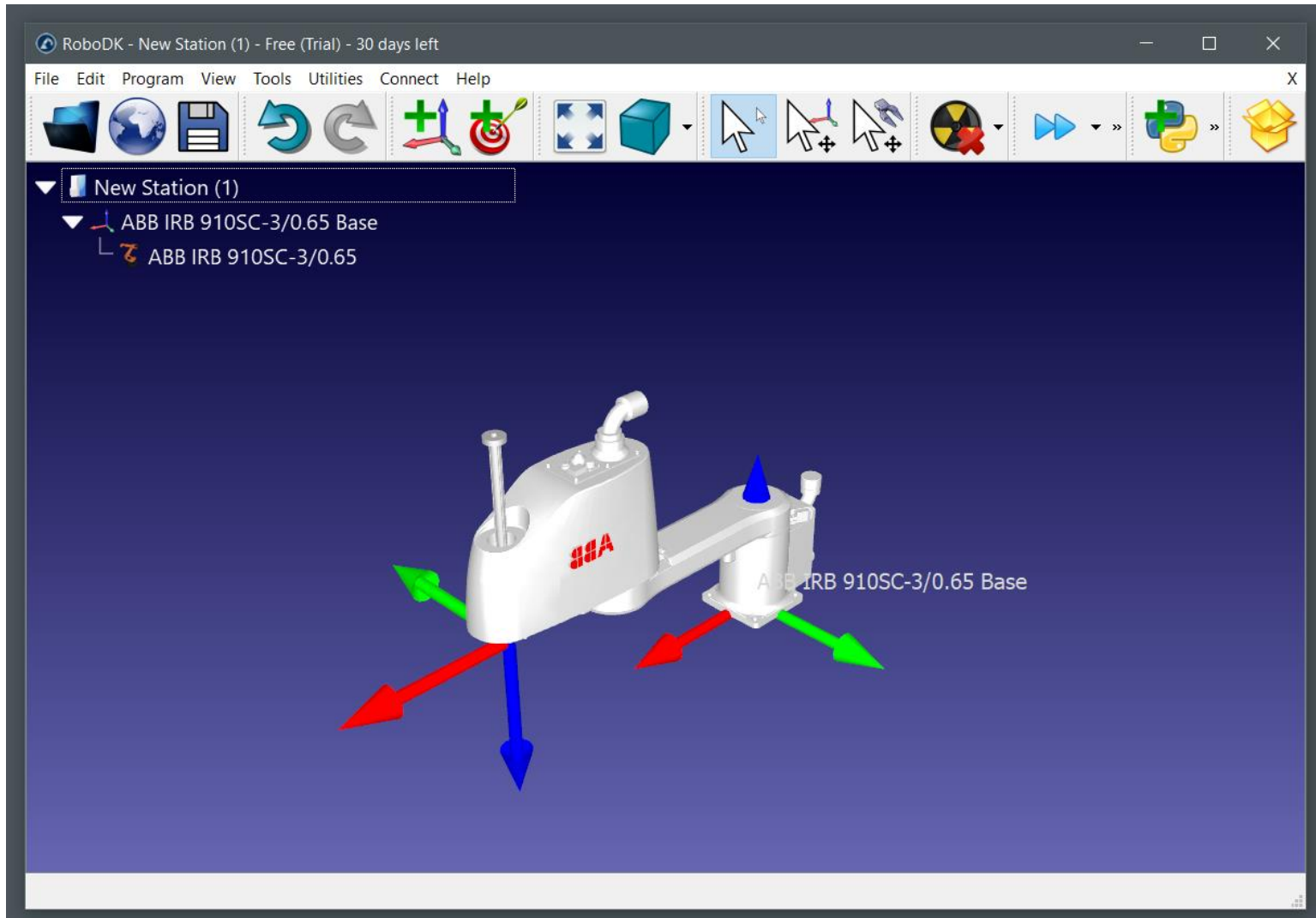
Robot Simulator

- RoboDK <https://robodk.com/>
 - Pros: 4-axis SCARA is available
 - Cons: 30-days free trial, limited to 50 lines of code
- Cyberbotics <https://cyberbotics.com/>
 - Pros: Open source
 - Cons: 4-axis SCARA is not available
- Code in Python
 - Pros: Numerous tutorials on internet
 - Cons: 3D-simulated robot is not available

Original Implementation



RoboDK



RoboDK

The screenshot displays the RoboDK software interface. The main window shows a 3D model of a KUKA KR 6 R900 sixx robot arm with a welding torch. The interface includes a menu bar (File, Edit, Program, View, Tools, Utilities, Connect, Help), a toolbar, and a left-hand tree view showing the project structure: Offline programming - W..., KUKA KR 6 R900 sixx B..., KUKA KR 6 R900 sixx, Weld gun, Home, Target 1, and DoWeld.

The Code Editor window displays a Python script named `DoWeld.py` with the following content:

```
1 from robolink import * # API to communicate with RoboDK
2 from robodk import * # robodk robotics toolbox
3
4 # Any interaction with RoboDK must be done through RDK:
5 RDK = Robolink()
6
7 # get the robot by name:
8 robot = RDK.Item('', ITEM_TYPE_ROBOT)
9
10 # get the home target and the welding targets:
11 home = RDK.Item('Home')
12 target = RDK.Item('Target 1')
13
14 # get the pose of the target (4x4 matrix representing position and orientat
15 poseref = target.Pose()
16
17 # move the robot to home, then to the Target 1:
18 robot.MoveJ(home)
19 robot.MoveJ(target)
20
21 # make an hexagon around the Target 1:
22 for i in range(7):
23     ang = i*2*pi/6 #angle: 0, 60, 120, ...
24     pose1 = poseref*rotz(ang)*transl(200,0,0)*rotz(-ang)
25     robot.MoveL(pose1)
26
27 # move back to the center, then home:
28 robot.MoveL(target)
29 robot.MoveJ(home)
30
```

The Options dialog box is open, showing the Python settings tab. The Python folder path is `C:/Python34`, the Python path is `C:/Python34/python`, and the Python editor command is `"C:/Program Files/Microsoft VS Code/Code.exe"`. The `Show program editor as an embedded window` checkbox is checked. A smaller dialog box is open over the Python editor command field, titled "Select the Python editor used by RoboDK", with the following options:

- "C:/Program Files/Microsoft VS Code/Code.exe"
- "C:/Python34/pythonw" "C:/Python34/Lib/idlelib"
- "C:/Program Files (x86)/Notepad++/Notepad++.exe"

The status bar at the bottom indicates "Python 3.7.2 32-bit" and "Ln 13, Col 1 Spaces: 4 UTF-8 LF Python".

<https://robodk.com/doc/en/PythonAPI/intro.html#windows>

