Module name: Robot control for bin picking

- **Main functionalities**:

Robot control for bin picking works as an integrator of object detection and classification modules or any other system which provides pickable object data. This module is mainly responsible for movement and trajectory creation depending on object position in the container, object class or other information from sensors. In the selected demonstration example for picking and placing of arbitrary arranged different objects, the robot movement is implemented in following way – linear path generation for picking up an object respectively to its position and orientation, while avoiding collisions with obstacles in the environment, linear path generation to move object in classification position to classify what kind of the object has been picked and other features of the object, after that the object is moved to appropriate position(box) depending on classification result. Operation flow is shown in Fig. 7.

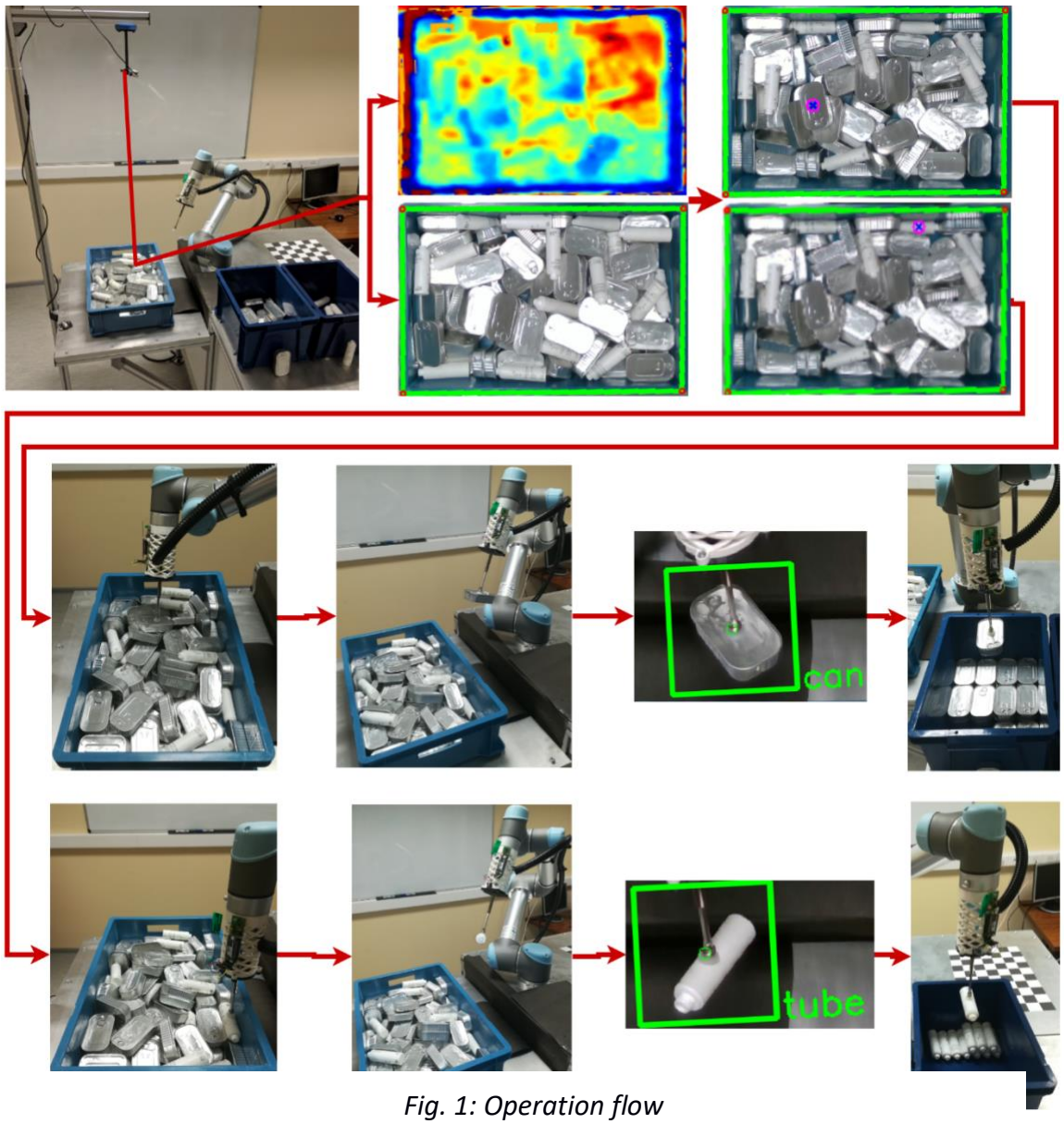

Fig. 1: Operation flow

- **Technical specifications:**

The overall system description is shown in Fig. 8. ROS Kinetic on Ubuntu 16.04 PC is used. ROS-I is being used in the module as a high-level controller in conjunction with a low-level controller provided by an industrial robot. Currently, Universal Robot UR5 is supported, but the module can be adjusted for other ROS compatible industrial robots. ur_modern_driver and universal_robot ROS packages are used to establish the connection between the Robot Control node and robot low-level controller.

Interface with MoveIt! layer is managed through a move_group_interface package where access to move_group node is realized with C++ API. As MoveIt! works with motion planners through a plugin interface it is possible to choose easily the most appropriate motion planner for corresponding automation task and regarding workspace. Currently, STOMP motion planner and Trace-Ik inverse kinematics solver are used.

Vacuum gripper is specially made for the experimental setup and strictly depends on automation task, objects and the workspace parameters. Gripper control is integrated into the Robot Control node. Robot control module accuracy strictly depends on the precision of detected object pose.

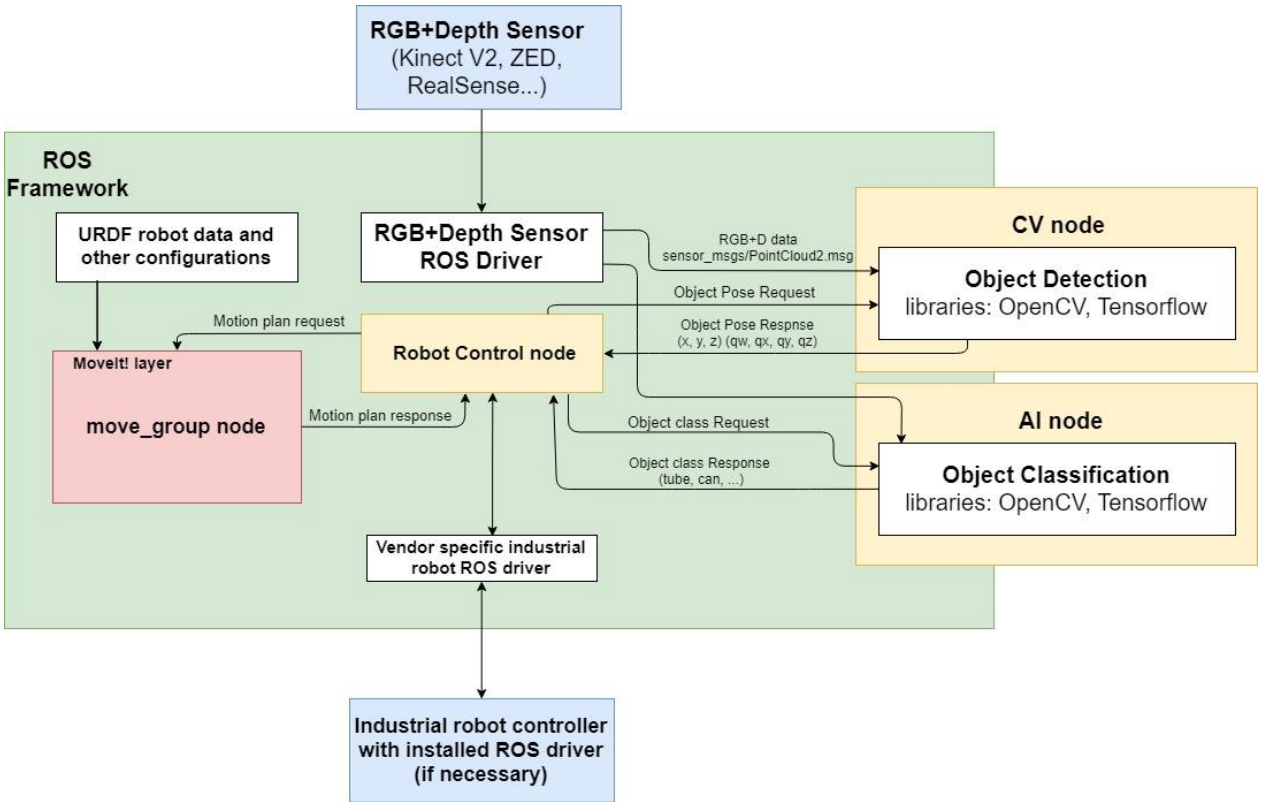

Fig. 2: Robot control module architecture

- **Inputs and outputs:**

All the data is transferred via a standard ROS transport system with publish/subscribe and request/response semantics.

This module takes object pose data: position (x, y, z) and orientation in quaternion format (qw, qx, qy, qz) from the object detector module to generate collision-free trajectory for object picking from a bin and placing the object depending on classification result.

- **Interface specification:**

Rviz for digital robot workspace and movements.

- **Formats and standards used**:

ROS-Industrial, MoveIt!, ROS communication layer.

- **Availability:**

Available, repository link can be provided by request.

- **Application scenarios:**

Bin-Picking

- **Offered for internal / external use**

Internal and external