**Module name**: Depth-sensor safety model for HRC

- **Main functionalities**:

*Depth-based safety model for human-robot collaboration. Generates three different spatial zones in the shared workspace which are then online modelled, updated and monitored.*

- **Technical specifications:**

The overall description of the hardware requirements and the different software nodes in the module are shown in Fig. 6. The workspace is monitored by the Kinect v2 sensor at the frame rate of 30 Hz and the robot is UR5 from the Universal Robot family. All the nodes exchange messages using the TCPROS transport layer where the nodes that are interested in data *subscribe* to the relevant topic and the nodes that generate data *publish* to the relevant topic. Arrows show the direction of the transmission.

A modified version of *ur_modern_driver* and *univeral_robot* ROS packages are used to establish communication channel between the robot low-level controller and the safety system node. Iai-kinect2 ROS package is used to receive data from the Kinect-2 sensor and further transmit it to the safety node which monitors safety violations and changes on the workspace.

The robot and depth sensor are connected to a single laptop computer that runs the ROS Melodic distribution on Ubuntu 18.04 and performs all computing. To successfully compile the module, OpenCV and PCL libraries must be installed in addition to standard C++ libraries. Currently Kinect v2 and Universal Robot 5 are supported.

- **Preliminary software configurations:**

Before the module can be used 1) the robot and sensor must be extrinsically calibrated, 2) size of the virtual safety hull has to be defined and 3) wired communication link between PC and robot must be established.

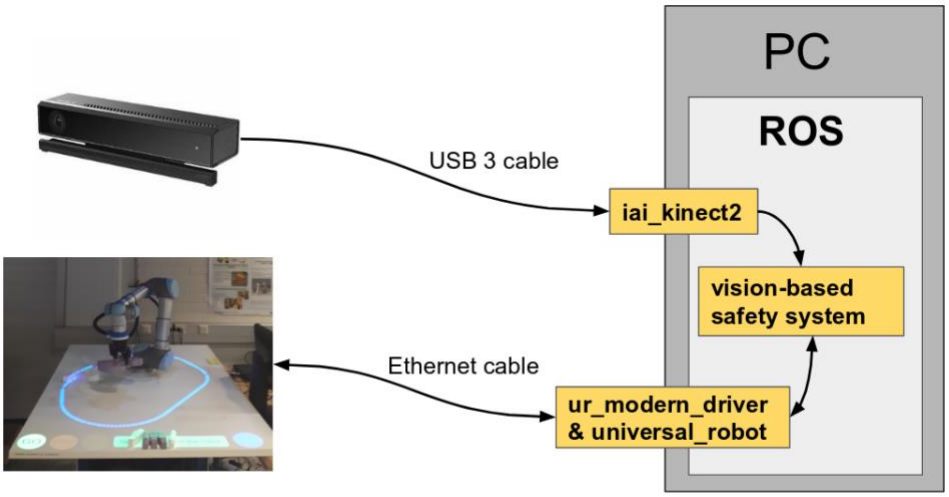


Fig. 1. Module software nodes and the hardware components.

- **Inputs and outputs:**

All the data is transferred via a standard ROS transport system with publish / subscribe semantics. Input and output data formats as well as the topic names are shown in Fig. 7 and Fig. 8. The vision-based safety system subscribes to topics where it can receive the color and depth image and the *CameraInfo* message which contains the sensor intrinsic parameters. In addition, the information from the *JointState* message is used to generate the safety hull.

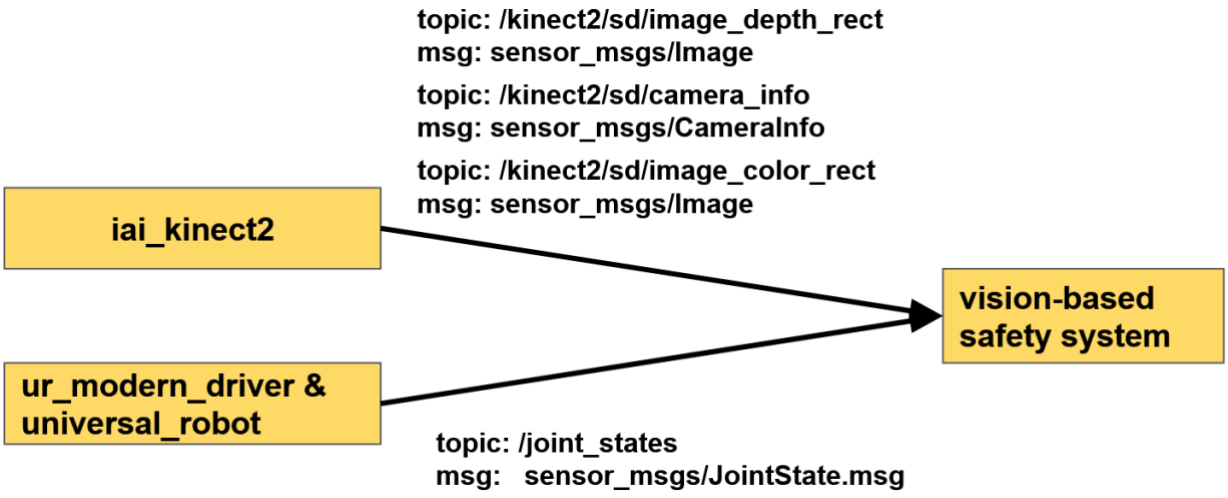


Fig. 2

The only output of the node is the stop command for the robot which is published over the */ur_driver/dashboard_command* topic.


Fig. 3

- **Interface specification:**

The end-user does not directly interact with the module. The module communicates with the robot controller and stops the robot if needed.

- **Formats and standards used**:

ROS communication layer with external *image_transport* package. Details about the message formats can be found from http://wiki.ros.org/sensor_msgs. In addition ROS-industrial, OpenCV, PCL and C++ and Python standard libraries.

- **Availability:**

Module library: https://github.com/Herrandy/HRC-TUNI.git

- **Application scenarios:**

Industrial assembly.

- **Offered for internal / external use**

Available for internal/external use.