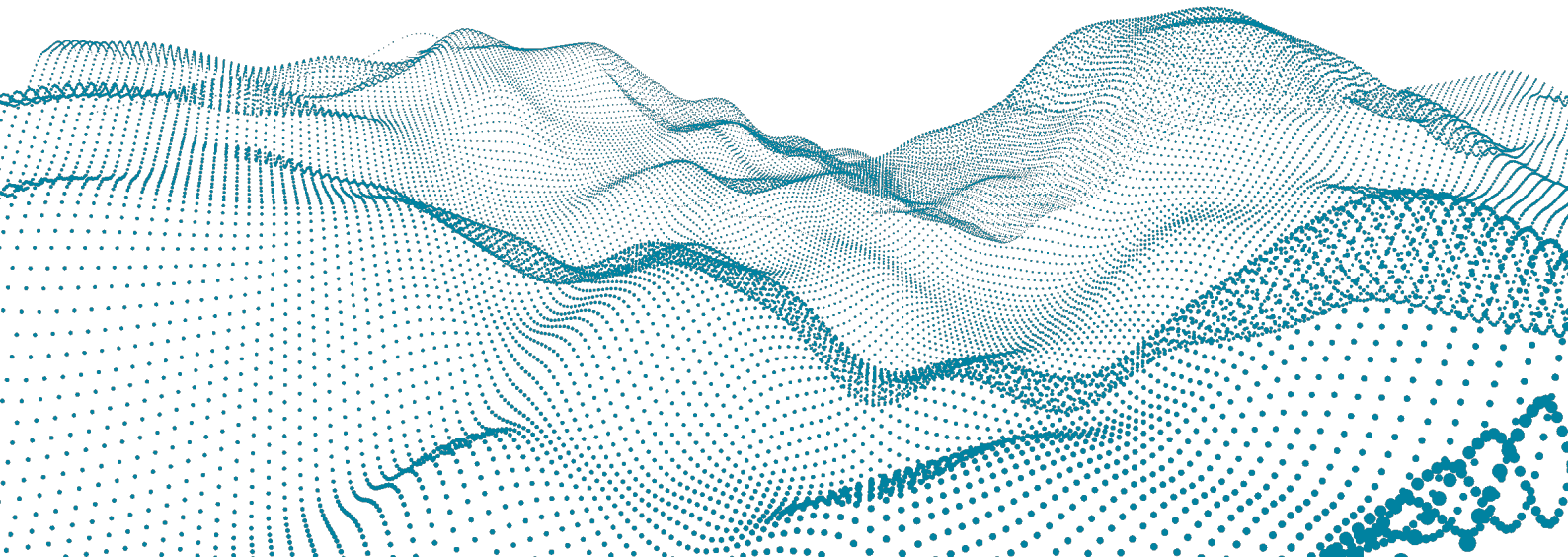
roboception

Roboception GmbH | October 2026

rc_reason_stack

3D Vision Software Platform

SOFTWARE CHANGELOG



Contents

1 26.10.0 (2026-10-26) 2
1.1 New Features . . . 2
1.2 Improvements and Fixes . . . 2

2 26.07.0 (2026-07-28) 3
2.1 New Features . . . 3
2.2 Improvements and Fixes . . . 4

3 26.04.2 (2026-06-22) 5
3.1 Improvements and Fixes . . . 5

4 26.04.1 (2026-05-22) 5
4.1 Improvements and Fixes . . . 5

5 26.04.0 (2026-04-28) 5
5.1 New Features . . . 5
5.2 Improvements and Fixes . . . 5

6 26.01.0 (2026-01-30) 6
6.1 New Features . . . 6
6.2 Improvements and Fixes . . . 7

1 26.10.0 (2026-10-26)

1.1 New Features

- Plane-based dimensioning in ItemPickAI (`rc_itempick`) and BoxPick (`rc_boxpick`): The new optional `dimensioning` argument of `compute_grasps` and `detect_items` measures a bounding box for each detected item that is aligned with a reference plane, e.g. a conveyor belt or a table. The box height is measured from that plane, so the full dimensions of the items are returned in addition to the grasps. The reference planes are taken from the new PrimitiveDB module (`rc_primitive_db`). This allows measuring the size of parcels, boxes and other items in the same call that computes the grasps, e.g. for sorting items by size or choosing their place position.
- PrimitiveDB (`rc_primitive_db`): This new database module stores geometric primitives, currently planes, persistently on the device, so that they can be used by the detection modules, e.g. as reference planes for dimensioning. Planes are managed with the new services `set_plane`, `get_planes` and `delete_planes`, or on the Web GUI's new Planes database page, which also helps to estimate a plane from the current sensor data. Thus, a reference plane only has to be determined once during setup and can then be reused in every detection request.
- Measure (`rc_measure`): The new service `compute_plane` estimates a plane from the current sensor data, either from the point cloud (`DEPTH`) or from at least three AprilTags lying on a common plane (`APRILTAG`). The plane is returned without being stored and can be stored in PrimitiveDB with `set_plane`. The Web GUI's Measure page shows the estimated plane in a 3D visualization, and the plane can be adjusted interactively. This makes it easy to determine reference planes such as a conveyor belt or a table surface directly with the sensor instead of measuring them manually.
- Stereo Matching (`rc_stereomatching`): The new parameter `method` selects the stereo matching method. Besides the classic `SGM`, the AI-based `FastStereoAI` method is available, which uses a neural network for computing the disparity image. It can deliver more complete depth data in scenes that are difficult for classic stereo matching, e.g. with weakly textured or reflective surfaces.
- Generic Robot Interface (GRI): The Web GUI's job editor now supports dynamic arguments, additional return values and mapping tables. Parameter jobs can be created, and jobs can be loaded and edited directly from the Try-Out sections of the modules. This makes setting up robot programs via GRI faster and less error-prone, as a configuration tested in the Try-Out section can be turned into a job directly.
- AprilTags (`rc_april_tag_detect`) and QR Codes (`rc_qr_code_detect`) can now be detected on all pipeline types. On blaze, Orbbec and Zivid pipelines the exact tag size has to be given for every tag in a detection request. This allows applications that rely on tag or code detection to be used with all supported sensors.

1.2 Improvements and Fixes

- Fixed CVEs
- Deployment/Docker compose:
  - Removed the `triton-model-init` service: rc_container now provides only the AI models that are covered by the available licenses to the inference server. Existing deployments need the updated compose files.
  - Updated the inference server image to `tritonserver:25.02-rc-20261008`
  - The inference server logs are now stored in `/data/triton/logs` and included in the downloadable log files
  - CodeMeter can access a USB license dongle directly if `/dev/bus/usb` is mounted into the container
- ItemPickAI (`rc_itempick`):
  - Improved quality of the "Mail Items" model
  - Faster grasp computation for partially covered items

  - Fixed rare cases when x axis of the item bounding box was not aligned with the long side
  - Fixed `discard_cut_off_items` on blaze, Orbbec and Zivid pipelines
- BoxPick (`rc_boxpick`):
  - Fixed `discard_cut_off_items` on blaze, Orbbec and Zivid pipelines
- Pack (`rc_pack`):
  - Added argument `robot_pose` to `detect_place_pose` for robot-mounted sensors
  - Fixed load carriers with a pose in the external frame
  - Fixed load carrier detection not respecting `assume_gravity_aligned`
- SilhouetteMatch (`rc_silhouettematch`):
  - Fixed objects close to the border of a 2D region of interest not being detected
- CADMatch (`rc_cadmatch`):
  - Improved GPU memory management
- GripperDB(`rc_gripper_db`):
  - Fixed uploading of CAD elements (OBJ files) with multiple connected components
- Generic Robot Interface (GRI):
  - Improved error handling with more specific return codes, e.g. for timeouts and misconfigured jobs
  - Poses and floating point values are now rounded instead of truncated
- Orbbec:
  - Added parameter `resolution` for choosing the resolution of the depth image
  - Faster image acquisition
- Zivid:
  - Updated Zivid SDK to 2.18.0
  - Fixed heartbeat timeouts when a continuous detection takes longer
- REST-API:
  - Reduced memory usage
  - Added timeouts so that a hanging module can no longer block REST-API requests
- Web GUI:
  - Added a popup for previewing the request and response JSON of a result, with options to copy it to the clipboard or download it
  - Added 3d visualization for `measure_depth` result to Measure page
  - Reduced the CPU load of the 3D result visualizations
  - Minor improvements and fixes

2 26.07.0 (2026-07-28)

2.1 New Features

- Pack (`rc_pack`): This new module computes placement poses for objects inside a packing volume, such as a bin or a pallet. It takes the current filling state of the packing volume into account and selects the lowest position that provides stable support for the object, thereby maximizing the remaining space after placement and keeping as many packing options open as possible for the next object to be placed.
- A new object category, “Mail Items”, has been added to ItemPickAI (`rc_itempick`), enabling detection of mail items such as parcels, packages, and shipping bags. Suction grasps are computed at the center of each object so that oriented placement is possible.

- ItemPickAI (`rc_itempick`) and BoxPick (`rc_boxpick`) now offer a new run-time parameter, `discard_cut_off_items`, which discards objects cut off by the image border, the stereo ROI, the 3D ROI, or the stereo overlap. This ensures that grasps are only computed for objects that are fully within the field of view.
- CADMatch (`rc_cadmatch`) offers a new service call for teaching grasp poses directly with the robot: move the robot into the desired grasping pose and record its TCP (Tool Center Point) pose. This makes grasp definition easier and more reliable, especially for complex grippers or grippers that contact the object at multiple distant points, where small rotational errors in a manually defined grasp point can have an outsized effect on the actual gripper pose.

2.2 Improvements and Fixes

- Fixed CVEs
- ItemPickAI (`rc_itempick`):
  - Improved quality of the “Sheet Metal” object segmentation model
  - Faster processing times due to inference speed-up
  - Added new service `detect_items`, which returns the items with their poses and bounding boxes
  - Fixed grasp orientation when the grasp’s long axis is not aligned with the object’s long axis
- SilhouetteMatchAI (`rc_silhouette`):
  - Improved quality of the “Sheet Metal” object segmentation model
  - Faster processing times due to inference speed-up
- BoxPick (`rc_boxpick`):
  - Items returned by `detect_items` are now also sorted according to the user-defined sorting strategy
- LoadCarrier (`rc_load_carrier`):
  - Added new service `detect_filling_level_as_image`, which returns filling level values as compact, zlib-compressed, base64-encoded image strings for faster responses when the number of cells is large
- GripperDB (`rc_gripper_db`):
  - Increased maximum element dimensions and position bounds from 1.0 to 5.0 meters
- All detection modules:
  - The return code value and message of the last detection are now reported in the modules’ status values
- Generic Robot Interface (GRI):
  - Support returning additional data in data_4 - data_10
  - Support passing dynamic arguments in data_1 - data_4
- Stereo ace:
  - Added support for cameras with firmware >= 1.5.0 to fix a synchronization bug
  - Added network-interface-based device filtering
- Web GUI:
  - Improvements to the blaze, Zivid, and Orbbec pipelines
  - Added orthographic view mode to 3D viewers
  - Minor improvements and fixes
  - Show uptime of container instead of host
- Deployment/Docker compose
  - Added heathcheck to rc_container and tritonserver

  - Ensure default gateway is set correctly
  - Add interface_name for sensor ports so these can be used to filter cameras by ethernet port
  - Parameters are now saved on shutdown to persistent volume, and restored when the container is re-created or updated

3 26.04.2 (2026-06-22)

3.1 Improvements and Fixes

- Fixed synchronization problem of Basler Stereo ace

4 26.04.1 (2026-05-22)

4.1 Improvements and Fixes

- Fixed CVEs
- Updated tritonserver to support NVIDIA Blackwell GPUs
- Stereomatching (`rc_stereomatching`): Fixed issues with camera triggering
- Generic Robot Interface: Added retry in case of 'Too many requests' error

5 26.04.0 (2026-04-28)

5.1 New Features

- Stereo Matching (`rc_stereomatching`) can now be restricted to a user-defined 2D region of interest. This reduces latency when only a part of the image is relevant for the application.
- The gripper definition (`rc_gripper_db`) and CollisionCheck (`rc_collision_check`) support multiple TCPs: Each gripper can now have multiple TCPs with different poses in prioritized order. During collision checking, all defined TCPs are checked for collisions, and the ID of the first collision-free TCP will be returned for each grasp in all rc_reason software modules. By setting the minimum suction surface dimensions for each TCP, only grasps that fulfil these requirements are considered during collision checking. This is especially helpful in ItemPick, ItemPickAI and BoxPick for applications with multi-suction-cup grippers, where different combinations of suction cups can be activated depending on the size of the item to grasp.
- CADMatch (`rc_cadmatch`) now allows to set the direction in which the object overlap should be checked and the overlap check is always performed orthographically. This gives more flexibility in applications where the camera perspective does not correctly capture the overlap situation, e.g. when the camera has a slightly tilted viewing angle or the perspective projection leads to wrong overlap detections.
- Support network licenses: If the CodeMeter runtime cannot be installed on the host, the environment variable `CM_REMOTE_SERVER` can be set to the IP address or name of a remote license server.
- Support arm64 devices: The rc_reason_stack can now also be deployed on arm64 devices, such as NVIDIA Jetson Orin.

5.2 Improvements and Fixes

- Added shared memory setting to docker-compose.yml to fully support rc_viscore cameras
- ItemPickAI (`rc_itempick`):
  - Updated the CONSUMER_GOODS model

  - Fixed missing grasps on detected items
  - Made z axis of item always point towards the camera to be consistent with the item orientation in BoxPick
  - Added warning code 115 when additional items without grasps were found
  - No longer return code 101 when no grasps are found, but valid clusters were detected
- CADMatch (`rc_cadmatch`):
  - Prior accessibility now also considers whether the object has reachable grasps
- CollisionCheck:
  - Fixed collision checking when FINGER elements have parent 'flange'
- SilhouetteMatch (`rc_silhouettematch`):
  - Added match numbers in result visualization
- EKI Bridge:
  - Updated the EKI XML config files in manual to utilize the maximum 64kB `BUFFSIZE`, allowing for more data per request
  - Added automatic payload pruning to prevent EKI connection crashes and transmission errors when responses would exceed the `BUFFSIZE` limit
- gRPC Interface:
  - Added gamma and auto_exposure_adapting to Image
- REST-API:
  - Added system messages to inform about the current system status
- Stereo ace:
  - Fixed regression in rc_reason_stack v26.01: Stereo ace cameras with firmware version 1.4.0 and above were not supported
- Web GUI:
  - Use Turbo color map in color-coded disparity image for better visibility of slight disparity changes
  - Fixed grasp duplication and downloading issues in CADMatch and SilhouetteMatch templates
  - Added warning bar when a license is about to expire
  - Minor improvements and fixes

6 26.01.0 (2026-01-30)

6.1 New Features

- SilhouetteMatchAI (`rc_silhouettematch`): This advancement of SilhouetteMatch allows to use AI models for segmenting planar objects in the scene, so that no base plane calibration is required. To use the AI-based segmentation, set `object_plane_detection` to true and select `SHEET_METAL` as `object_segmentation_model` in the detection service. In addition, when the AI-based segmentation is used, the new parameter `max_object_overlap` allows to specify a maximum fraction of the object that can be overlapped by other segmented objects. Objects with higher overlap ratios will be discarded. Thus, only grasps on unoccluded matches are returned. These new features allow for safely and precisely picking partly occluded metal parts. The SilhouetteMatchAI module requires a separate license.
- SilhouetteMatch (`rc_silhouettematch`) is now also available for Zivid and Orbbec camera pipelines. This way SilhouetteMatch can make use of the Zivid's highly accurate point clouds for precisely picking planar metal parts with challenging surfaces.

- ItemPickAI (`rc_itempick`): A new object category “Sheet Metal” is available. It can be selected by setting the `type` of the `item_model` to `SHEET_METAL`. The sheet metal model enables picking and oriented placing of flat planar metal objects of unknown size and geometry.
- Support of suction cups and movable fingers in gripper definition (`rc_gripper_db`) and CollisionCheck (`rc_collision_check`): When modelling a gripper, single gripper elements can be assigned a `function_type`, e.g. `SUCTION_CUP` or `FINGER`. Elements marked as suction cups will be ignored during collision checking, so that suction cups can now be modelled correctly for point cloud collision checking of suction grasps. Elements marked as fingers can have variable positions. By defining the zero position of the finger elements, variable strokes can be assigned to user-defined grasps in SilhouetteMatch and CADMatch. This makes the grasp definition with collision checking in complex bin picking applications more comfortable.
- New detection services for easier response parsing in ItemPickAI (`rc_itempick`), BoxPick (`rc_boxpick`), SilhouetteMatch (`rc_silhouettematch`) and CADMatch (`rc_cadmatch`): The new services `compute_grasps_extended` in ItemPickAI and BoxPick, and `detect_object_extended` in SilhouetteMatch and CADMatch perform the same detection tasks as their non-extended counterparts, but deliver the results in a different structure that is easier to parse. Instead of returning items or matches as a separate list next to the grasps, each grasp in the response now contains information about its corresponding object, e.g. the object’s pose that is required for exact placement. Thus, retrieving the object information has become much easier.
- gRPC Interface: A new `ImageEventsInterface` service allows sending `depth_acquisition_done` events that signal when the image acquisition for depth computation is completed. This event can be used to determine when it is safe to move the robot or change the scene while a detection is still in progress. This helps for optimizing cycle times in robot applications.

6.2 Improvements and Fixes

- CollisionCheck (`rc_collision_check`):
  - Added new parameter `pointcloud_watertight` that allows to switch off filling holes in the depth image for point cloud collision checking
- LoadCarrier (`rc_load_carrier`):
  - Increased the maximum values of `filling_level_cell_count` in `detect_filling_level` to 200
- added `trigger_dump` service calls in LoadCarrier, TagDetect, ItemPick, ItemPickAI, BoxPick, SilhouetteMatch and CADMatch
- Generic Robot Interface:
  - Support of fragmented TCP messages by TCP message buffering
- CADMatch (`rc_cadmatch`):
  - Improved refinement run-time
  - Added support for new templates with higher detection robustness
- ItemPickAI and ItemPick (`rc_itempick`):
  - Fixed wrong number of items returned if `allow_any_grasp_pose` is active
  - Removed empty `template_id` and `view` fields from ItemPick responses
- Hand-Eye Calibration (`rc_hand_eye_calibration`):
  - Prevent changing the stereo calibration in case of high hand-eye calibration errors
- Web GUI:
  - New CADMatch pose priors are added with flipped orientation by default
  - Minor fixes and layout improvements

roboception

rc_reason_stack

3D Vision Software Platform

SOFTWARE CHANGELOG

Roboception GmbH

Kaflerstrasse 2
81241 Munich
Germany

info@roboception.de
www.roboception.com

Tutorials: https://tutorials.roboception.com
GitHub: https://github.com/roboception
Documentation: https://doc.rc-reason.com
https://doc.rc-visard.com
https://doc.rc-cube.com
https://doc.rc-viscore.com
https://doc.rc-randomdot.com
Shop: https://roboception.com/shop

For customer support, contact

+49 89 889 50 790
(09:00-17:00 CET)

support@roboception.de

