indigo | kinetic *Show EOL distros:*

Documentation Status

Package Links

- **Code API (http://docs.ros.org/kinetic/api/depthimage_to_laserscan/html)**
- FAQ (http://answers.ros.org/questions/scope:all/sort:activity-desc/tags:depthimage_to_laserscan/page:1/)
- Changelog (http://docs.ros.org/kinetic/changelogs/depthimage_to_laserscan/changelog.html)
- Change List (/depthimage_to_laserscan/ChangeList)
- Reviews (/depthimage_to_laserscan/Reviews)

Dependencies (7)
Jenkins jobs (12)

Package Summary

✔ **Released** ✔ **Continuous Integration** ✔ **Documented**

depthimage_to_laserscan

- Maintainer status: maintained
- Maintainer: Chad Rockey <chadrockey AT gmail DOT com>
- Author: Chad Rockey
- License: BSD
- Bug / feature tracker: https://github.com/ros-perception/depthimage_to_laserscan/issues (https://github.com/ros-perception/depthimage_to_laserscan/issues)
- Source: git https://github.com/ros-perception/depthimage_to_laserscan.git (https://github.com/ros-perception/depthimage_to_laserscan) (branch: indigo-devel)

Inhaltsverzeichnis

1. Overview / Example Scene
    1. RGB
    2. DepthImage
    3. LaserScan
    4. Top Down LaserScan
2. Node
    1. depthimage_to_laserscan
        1. Subscribed Topics
        2. Published Topics
        3. Parameters
3. Nodelet

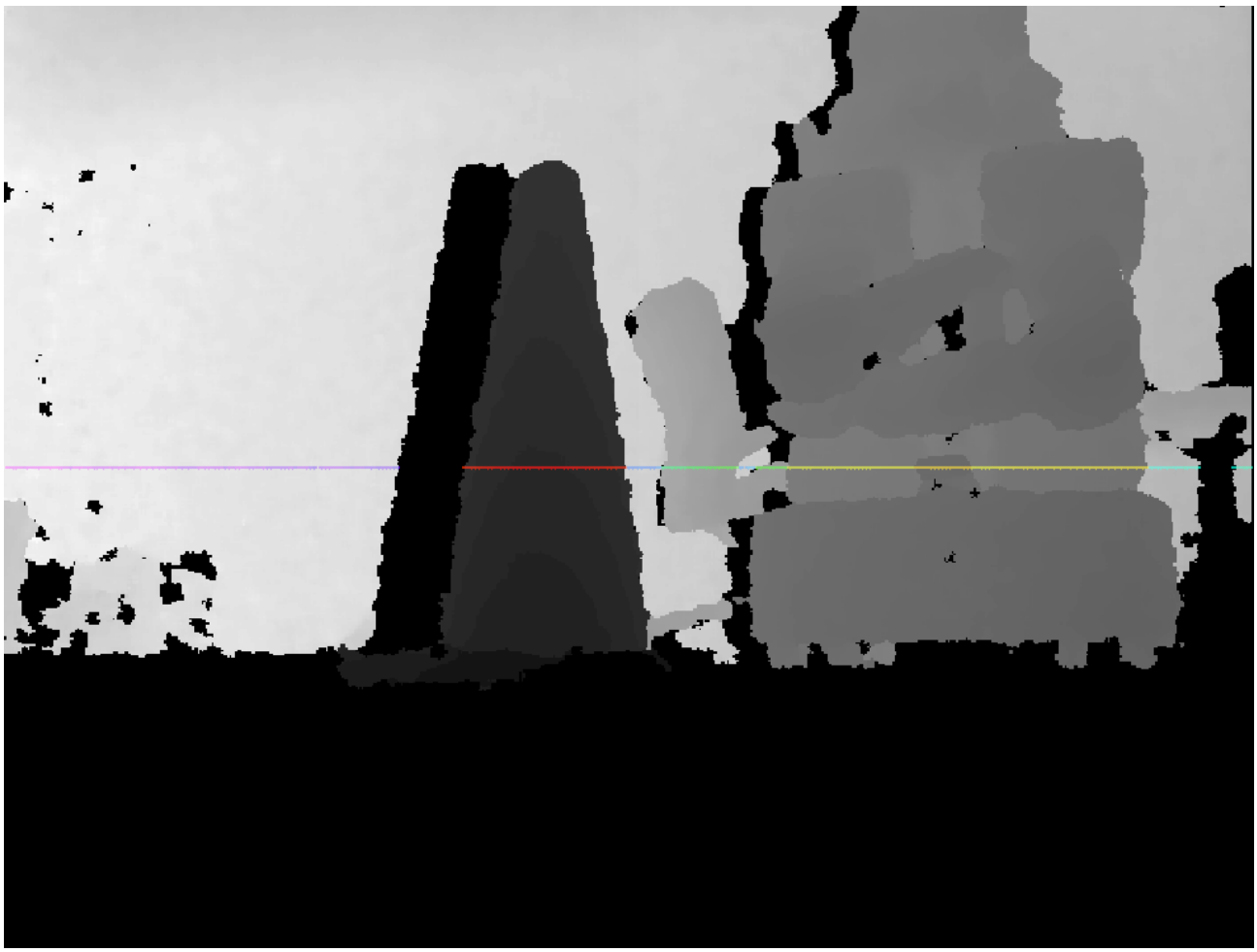
1. Overview / Example Scene

1.1 RGB

Here is the scene in which the following screenshots were captured.

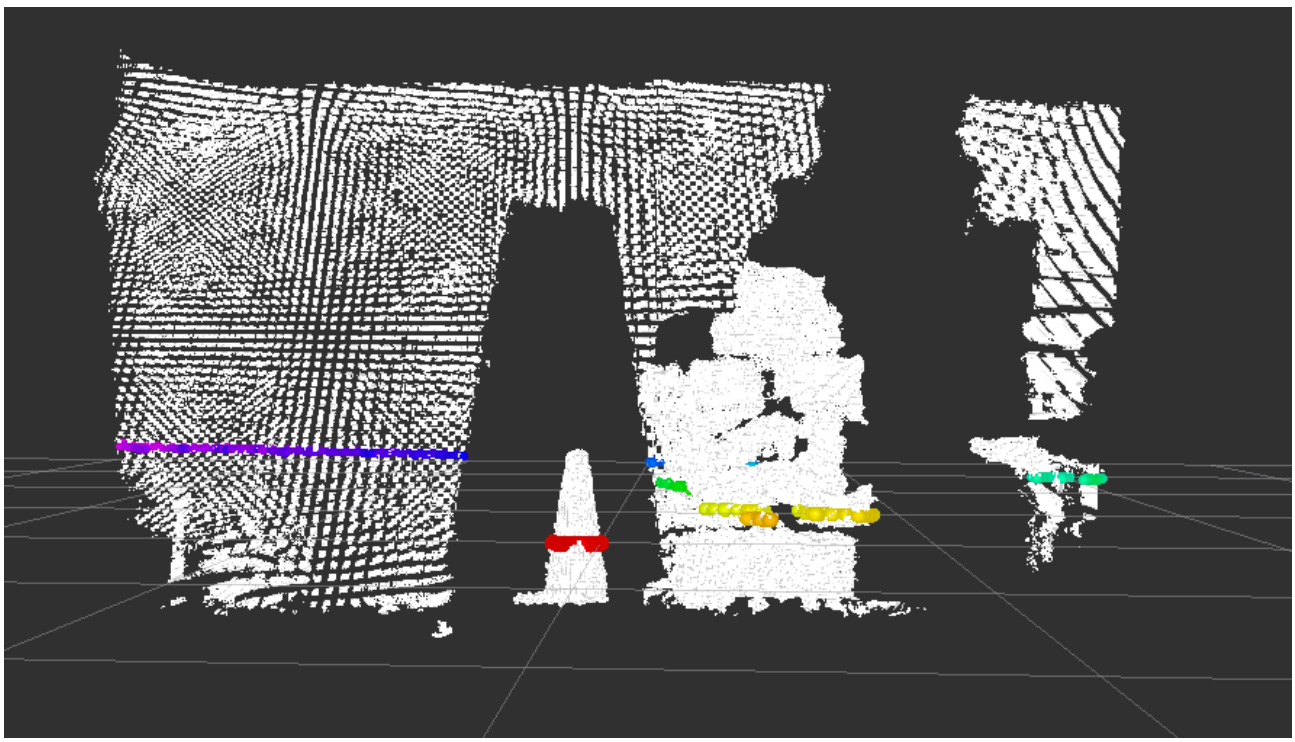
1.2 DepthImage

Note the sensor_msgs/LaserScan (http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html) overlayed in color on the sensor_msgs/Image (http://docs.ros.org/api/sensor_msgs/html/msg/Image.html). Red is close to camera, purple is far from camera.



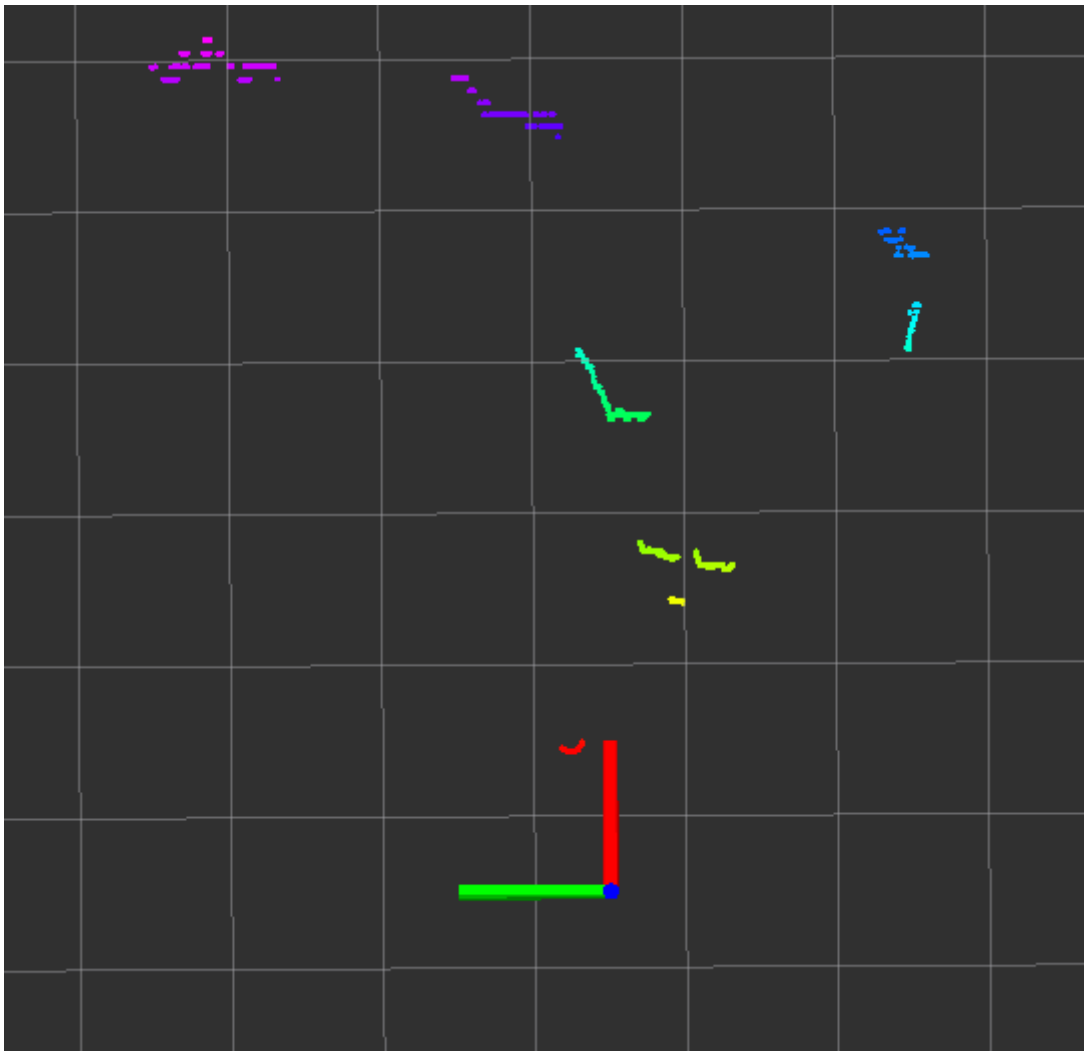
1.3 LaserScan

sensor_msgs/LaserScan (http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html) projected on top of the sensor_msgs/PointCloud2 (http://docs.ros.org/api/sensor_msgs/html/msg/PointCloud2.html).



1.4 Top Down LaserScan

Top down view of the sensor_msgs/LaserScan (http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html).



2. Node

2.1 depthimage_to_laserscan

`depthimage_to_laserscan` takes a depth image (float encoded meters or preferably uint16 encoded millimeters for OpenNI devices) and generates a 2D laser scan based on the provided parameters. depthimage_to_laserscan uses lazy subscribing and will not subscribe to `image` or `camera_info` until there is a subscriber for `scan`.

2.1.1 Subscribed Topics

`image` (sensor_msgs/Image (http://docs.ros.org/api/sensor_msgs/html/msg/Image.html))

The input image that must conform to REP 118 (http://ros.org/reps/rep-0118.html). This can be floating point or raw uint16 format. For OpenNI devices, uint16 is the native representation and will be more efficient for processing. This is typically `/camera/depth/image_raw`. If your image is distorted, this topic should be remapped to `image_rect`. OpenNI cameras typically have little distortion and rectification can be skipped for this application.

`camera_info` (sensor_msgs/CameraInfo (http://docs.ros.org/api/sensor_msgs/html/msg/CameraInfo.html))
Camera info for the associated image. Does not usually need to be remampped as camera_info will be subscribed to from the same namespace as `image`.

2.1.2 Published Topics

`scan` (sensor_msgs/LaserScan (http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html))
The output laser scan. Follows REP 117 (http://ros.org/reps/rep-0117.html), and will output range arrays that contain NaNs and +-Infs.

2.1.3 Parameters

`~scan_height` (`int`, default: 1 pixel)
The number of pixel rows to use to generate the laserscan. For each column, the scan will return the minimum value for those pixels centered vertically in the image.

`~scan_time` (`double`, default: 1/30.0Hz (0.033s))
Time between scans (seconds). Typically, 1.0/frame_rate. This value is not easily calculated from consecutive messages, and is thus left to the user to set correctly.

`~range_min` (`double`, default: 0.45m)
The minimum ranges to return in meters. Ranges less than this will be output as -Inf.

`~range_max` (`double`, default: 10.0m)
The maximum ranges to return in meters. Ranges greater than this will be output as +Inf.

`~output_frame_id` (`str`, default: camera_depth_frame)
The frame id of the laser scan. For point clouds coming from an "optical" frame with Z forward, this value should be set to the corresponding frame with X forward and Z up.

3. Nodelet

Same usage as the Node.

Available as:

`depthimage_to_laserscan/DepthImageToLaserScanNodelet`

Wiki: depthimage_to_laserscan (zuletzt geändert am 2018-01-16 08:58:51 durch NickLamprianidis (/NickLamprianidis))