



Kinect for Xbox 360 is a low-cost controller free device originally designed for gaming and entertainment experience by Microsoft Corporation. This device is equipped with one IR camera, one color camera and one IR projector to produce images with voxels (depth pixels).

There's a three-axis MEMS accelerometer onboard ostensibly for determining the tilt of the sensor relative to the base of the device

The most prominent physical feature on the Kinect, however, is its relatively sophisticated optical system. From left to right is an IR laser projector, RGB camera, and IR camera. Between the RGB camera and IR laser is simply a green LED for status - it plays no role in the actual optical system. Also on the bottom of the sensor is a nondescript class-1 laser product notice - yes, the Kinect indeed uses an IR laser, but it's completely eye safe

In Kinect, that system is comprised of an IR laser, a carefully engineered diffraction grating (in this case, the diffraction grating is actually a computer-generated hologram - CGH - with a specific periodic structure), and a relatively standard CMOS detector with a band-pass filter centered at the IR laser wavelength. Inspecting the sensor with the naked eye, you can easily see the characteristic rainbow-effect

Here we are using RVIZ as a platform to visualize 3D plotted data from IR of kinect in Rviz with laserscan as datatype.

Steps to install and configure XBOX 360 on Ubuntu(18.04):

NOTE: Open NI platform should not be used as it no longer works with ROS Melodic. Updated version of Open NI is Freenect which has to be used.

Steps to install freenect for kinect interfacing:

1. `sudo apt-get install libfreenect-dev`

2. `sudo apt-get install ros-melodic-freenect-launch`

3. `git clone https://github.com/ros-drivers/freenect_stack.git`

after this step you must check if all .hpp , .cpp and .h files are included in the package freenect in your workspace

4. `roslaunch freenect_launch freenect.launch depth_registration:=true`

5. `roslaunch freenect_launch freenect.launch`

Install RTABMAP\_ROS

1. `sudo apt-get install ros-melodic-rtabmap-ros`

2. `roslaunch rtabmap_ros rgdb_mapping rgdb_odometry:=true`

3. install additionally rgbd files from ros.org

Install depthimage\_to\_laserscan

1. `sudo apt-get install ros-melodic-depthimage-to-laserscan`

2. `roslaunch depthimage_to_laserscan depthimage_to_laserscan image:=/camera/depth/image_raw`

Install gmapping

1. `sudo apt-get install ros-melodic-slam-gmapping`

2. `roslaunch gmapping slam_gmapping scan=base_scan`

## CREATING A 2D SCAN FROM THE KINECT

1. `roscore`

2. `roslaunch freenect_launch freenect.launch`

3. `roslaunch depthimage_to_laserscan depthimage_to_laserscan`

4. `rviz`

## TO EXTRACT AND INTERPRET DATA FROM LASERSCAN

basically we have to run some python files to get going with the data

laserscan plots the obstacle in front of it in form of data points

firstly there is a need to find the number of data points in the particular laserscan line  
to find it run a python file making a new package in the workspace

after finding the total number of data points and the range angle of the kinect sensor logic for obstacle avoidance like when to go straight or to turn right has to be applied by making a python file in it.