



Intel® RealSense™ Camera

Depth Testing Methodology

Jan 2018

Revision 1.0



You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at Intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit www.intel.com/design/literature.htm.

By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel, RealSense, and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2017, Intel Corporation. All rights reserved.

Contents

1	Technical Overview	5
1.1	Terms and Acronyms.....	5
1.2	Introduction	5
1.3	Metrics Definitions	7
1.3.1	Z-Accuracy	7
1.3.2	Fill Rate.....	8
1.3.3	RMS Error (Spatial Noise)	9
1.3.4	Temporal Noise	10
2	Depth Quality testing for D400 cameras	12
2.1	Introduction to D400 family	12
2.2	Depth Quality Tool Introduction	13
2.3	Target Considerations.....	14
2.4	Setup	14
2.4.1	Target Distance	14
2.4.2	Lighting.....	14
2.5	Depth Quality Tool Operation	14
2.5.1	Reporting	16
2.6	Subjective Testing Considerations	16
2.6.1	Lighting.....	16
2.6.2	Materials	17
2.6.3	Distance	18
2.6.4	Shape	18

Revision History

Revision	Description	Date
1.0	<ul style="list-style-type: none">Initial Version	Jan 2018

§ §

1 Technical Overview

1.1 Terms and Acronyms

The table below lists terms and acronyms that appear in this document.

Term	Description
DQT	Depth Quality Tool
ROI	Region of Interest
SL	Structured Light based Depth Technology
TOF	Time of Flight based Depth Technology
FOV	Field of View
RMS	Root of Mean Square
SDK	Software Development Kit

1.2 Introduction

There are a variety of metrics that may be defined to evaluate depth quality. Described here are the basic metrics from which others may be derived or may be formed from combinations. The minimum requirements for measuring each of the basic metrics are different but it is recommended that a common environment be used that is compatible with all tests. This can be described generally as a flat surface that is placed at a controllable and known distance from the camera under test. Ideally, the target surface is smooth with a uniform, non-glossy (diffuse) finish yet with high reflectivity. A flat wall painted with a flat (or matte) white paint is a simple example of an acceptable test target. Alternatively, white poster board mounted to a rigid frame or structure may also be a suitable target.

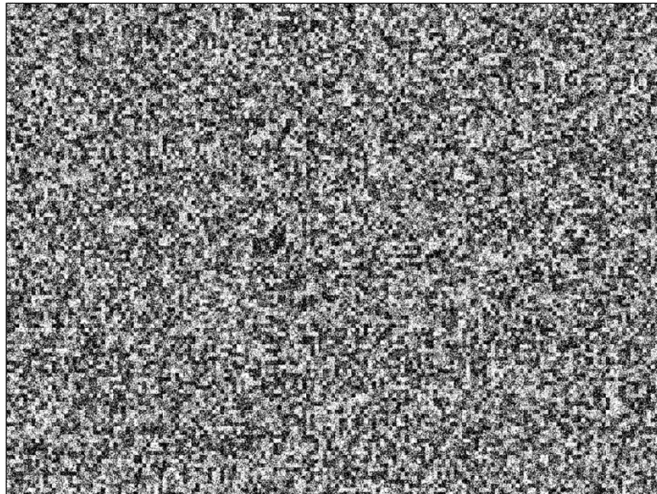
In cases where the camera has no internal projector, or the projector is turned off, a 'passive' test is performed, in which case the flat target must also provide a sufficiently textured pattern. In this case, special textured printed targets are used or projected onto the white, 'active', target surface. Sufficient ambient lighting is also required when performing passive tests, but this should not require more than typical room lighting.

Examples of suitable targets are shown below. Note that the 'active' target can only be used with active depth cameras whose projector provides the required texture, whereas the 'passive' target can be used in both active and passive camera modes.

Figure1-1: Texture-less (white) target or flat wall for active tests



Figure1-2: Textured target for passive tests



Depth metrics are typically measured in one of two ways: a) by capturing depth images from the camera and then analyzing the data as a separate, typically off-line, process; or b) performing the image capture and metric analysis in real time using a dedicated tool designed for depth quality measurements. The Depth Quality Tool provided by Intel is an example of a tool that is capable of computing the basic metrics for RealSense D400 depth cameras in real time as well as capturing images for off-line analysis.

1.3 Metrics Definitions

The primary characteristics of depth camera image quality can be described by: depth data accuracy, depth data coverage, and its spatial and temporal noise (or uniformity). Each metric may be computed over any desired ROI within the camera's FOV. Collectively, these metrics describe the primary depth-related performance of the camera, yet are each independent of the others

- **Z-accuracy** evaluates the depth data accuracy
- **Fill Rate** evaluate the percentage of the depth coverage of the image
- **RMS error** evaluates the spatial noise or spatial depth uniformity
- **Temporal Noise** evaluates the temporal uniformity over sequential frames

1.3.1 Z-Accuracy

Z-accuracy (or absolute error): Measures how close the reported depth values are to the actual distance (ground truth, GT). Z accuracy may be defined in a variety of ways but most definitions use differences (which may be signed or magnitude) between measured depth values (per pixel or averaged over an ROI) and GT. In most cases, a flat surface (target) aligned parallel to the depth camera is used in order to simplify the measurement since GT may be accurately estimated from a single measurement using an independent absolute measuring device such as laser rangefinder or tape measure.

Table 1-2-1 Z-Accuracy (Signed)

Item	Description
Category	Depth accuracy
Purpose/Goal	Measuring the accuracy of the depth image relative to GT
Target	wall \ Textured wall
Analyzed Image	N Vertices images
ROI	All Image
Description	<p>The metric measures the per pixel depth accuracy relative to GT (Image minus GT) for each of the N captured depth images, relative to the best fitted plane (isolating camera positioning errors). This yields a depth error map/matrix.</p> <p>Z-Accuracy (signed) = Median(Image – GT), i.e., the median value of the signed depth error map.</p> <p>An average value of the N results is then calculated.</p>
Pseudo-Code	<pre> LOOP for each of the N Vertices Images INPUT: Vertices_Image (X,Y,Z matrix), GT REMOVE all zeros of Z and all NaN in X and Y REMOVE outliers: all Z values that below percentile 0.5% and above percentile 99.5% COMPUTE Fitted_Plane to Vertices_Image COMPUTE Fitted_Plane_Normal COMPUTE Horizontal_tilt between [Y,Z] components of Fitted_Plane to [0,-1] of the target normal </pre>

Item	Description
	COMPUTE <i>Vertical_tilt</i> between [X,Z] components of <i>Fitted_Plane</i> to [0,-1] of the target normal COMPUTE <i>Rotated_Vertices</i> based on <i>Horizontal_tilt</i> (Ry) and <i>Vertical_tilt</i> (Rx) ASSIGN <i>Depth_Rotated</i> as Z comp. of <i>Rotated_Vertices</i> COMPUTE <i>Depth_Error</i> = <i>Depth_Rotated</i> – GT COMPUTE <i>Median_Value</i> = 50% percentile value of <i>Depth_Error</i> STORE <i>Median</i> value in <i>Medians_Array</i> COMPUTE <i>MedAvr</i> = average of <i>Medians_Array</i> OUTPUT: <i>Z-Accuracy_Signed</i> = 100* <i>MedAvr</i> / GT
Output Value	The average of the N described values, divided by GT
Output units	[%] of the GT

1.3.2 Fill Rate

Fill rate measures the portion (e.g., %) of the depth image containing valid pixels (pixels with non-zero depth values). This metric is independent of the accuracy of the depth data or its noise

Table 1-2-2 Fill Rate

Item	Description
Category	Depth coverage
Purpose/Goal	Measuring the percentile of the valid pixels from the total pixels within a pre-defined FOV
Target	wall \ Textured wall
Analyzed Image	N Depth images
ROI	Per selection (e.g., 100%, 80%)
Description	Calculating the percentile of pixels that have valid depth value, relative to the total pixels within the ROI
Pseudo-Code	LOOP for each of the N Depth Images INPUT: <i>Depth_Image</i> COMPUTE <i>Total_Count</i> = size (<i>Depth_Image</i>) ASSIGN <i>Valid_Image</i> = <i>Depth_Image</i> REMOVE all zeros from <i>Valid_Image</i> COMPUTE <i>Valid_Count</i> = size (<i>Valid_Image</i>) COMPUTE <i>Fill_Rate</i> = 100* <i>Valid_Count</i> / <i>Total_Count</i> STORE <i>Fill_Rate</i> value in <i>FR_Array</i> COMPUTE <i>Fill_Rate_Avr</i> = average of <i>FR_Array</i> OUTPUT: <i>Fill_Rate_Avr</i>
Output Value	The percentile of valid pixels within the ROI from the total pixels in ROI
Output units	[%]

1.3.3 RMS Error (Spatial Noise)

RMS Error (or Spatial Noise) measures the intrinsic variation in depth values across an ROI assuming a planar target. This metric is independent of the accuracy of the depth values or fill rate since invalid pixels are not included. A flat target is used to measure the spatial noise and it is typically defined as an RMS deviation of depth values from the best-fit plane to the data. Plane fitting of the data aids in reducing the sensitivity of the measurement to alignment errors between the camera and the target. A related measure of spatial noise is a standard deviation (STD) of the directly measured depth values. While STD can also provide a measure of spatial noise, any variation in depth across the ROI due to misalignment will bias the result and lead to a larger spatial noise value than is intrinsic to the camera.

Spatial noise may be quantified in either distance (e.g., mm) or disparity (e.g., pixels) units. Spatial noise in pixels (the fundamental depth camera unit), referred to as 'subpixel RMS error', has the advantage of being virtually independent of distance to the target, whereas noise measured in distance units is depth dependent but more commonly understood and relevant for most camera usages. For these reasons, both are typically used to evaluate camera performance.

Table 1-2-3 Plane Fit RMS Error

Item	Description
Category	Depth spatial noise
Purpose/Goal	Measure the depth spatial noise relative to the best fit plane.
Target	wall \ Textured wall
Analyzed Image	Vertices
ROI	All Image
Description	The metric calculates the RMS Error of each valid pixel from the best fit plane in mm.
Pseudo-Code	INPUT: <i>Vertices_Image (X,Y,Z matrix)</i> REMOVE all zeros of Z and all NaN in X and Y REMOVE outliers: all Z values that below percentile 0.5% and above percentile 99.5% COMPUTE <i>Fitted_Plane</i> to <i>Vertices_Image</i> COMPUTE $Z_Error = Original_Z - Fitted_Z$ COMPUTE $RMSE = \text{RootMeanSquare of } Z_Error$ OUTPUT: <i>RMSE</i>
Output Value	RMS Error from best fitted plane in mm.
Output Units	[mm]

Table 1-2-4 Subpixel RMS Error (pixels)

Item	Description
Category	Depth spatial noise
Purpose/Goal	Measure the depth spatial noise relative to the best fit plane.
Target	wall \ Textured wall
Analyzed Image	Vertices
ROI	All Image

Item	Description
Description	<p>The metric calculates the RMS Error of each valid pixel from the best fit plane in pixels.</p> <p>The calibration parameters provide the stereoscopic Baseline (the distance between the imagers) and the Focal-length as a multiple of pixel in the ROI (pixel units)</p> <p>The RMS is calculated on the difference between the disparity value in the ROI and the disparity value of the projected plane, both in pixels.</p>
Pseudo-Code	<p>INPUT: <i>Vertices_Image (X,Y,Z matrix), Calibration_parameters</i></p> <p>REMOVE all zeros of Z and all NaN in X and Y</p> <p>REMOVE outliers: all Z values that below percentile 0.5% and above percentile 99.5%</p> <p>COMPUTE <i>Fitted_Plane</i> to <i>Vertices_Image</i></p> <p>ASSIGN $BL = \text{Stereoscopic_Baseline}$; $FL = \text{Focal_length}$</p> <p>COMPUTE $Disparity = (BL \times FL) / \text{Original_Z}$</p> <p>COMPUTE $Disparity_Fitted = (BL \times FL) / \text{Fitted_Z}$</p> <p>COMPUTE $Disparity_Error = Disparity - Disparity_Fitted$</p> <p>COMPUTE $SubPixel_RMSE = \text{RootMeanSquare of } Disparity_Error$</p> <p>OUTPUT: <i>SubPixel_RMSE</i></p>
Output Value	RMS Error from best fit plane in pixels.
Output Units	[pixel]

1.3.4 Temporal Noise

Temporal Noise measures the variation in depth values over time. Temporal noise may be defined as a standard deviation in depth values over a specific number of frames (e.g., 10). It is typically measured per pixel and then an average or median is taken over the ROI to represent the temporal noise for that device and ROI.

Table 1-2-4 Temporal Noise

Item	Description
Category	Depth temporal noise
Purpose/Goal	Pixel Temporal noise measure time-dependent aspects of per pixel Z-accuracy.
Target	wall \ Textured wall
Analyzed Image	N sequential depth images
ROI	All Image
Description	<p>Pixel temporal noise measures the STD of the depth value per pixel (x,y) in N temporal images yield a STD matrix.</p> <p>Temporal noise is defined as the median value of the temporal noise of the entire image.</p> <p>The metric ignores nan/invalid depth data.</p>

Item	Description
Pseudo-Code	<p>INPUT: N images of <i>Depth_Image</i></p> <p>LOOP for each of the N <i>Depth_Image</i> to create <i>Depth_Tensor</i></p> <p> ASSIGN <i>Depth_Tensor</i> = (<i>Depth_Image</i>,N) //dimension is (x,y,N)</p> <p>REMOVE all zeros from <i>Depth_Tensor</i></p> <p>COMPUTE <i>STD_Matrix</i> = STD of <i>Depth_Tensor</i>(x, y, all)</p> <p>COMPUTE <i>Median_STD</i> = 50% percentile value of <i>STD_Matrix</i></p> <p>OUTPUT: <i>Median_STD</i></p>
Output Value	The median value of the STD matrix
Output units	[mm]

2 *Depth Quality testing for D400 cameras*

2.1 Introduction to D400 family

Intel® RealSense™ D400 family consists of a suite of cameras and modules that cater to a wide range of design choices ranging from baselines, FOVs and shutter types.

RealSense™ Depth Camera offers two versions: D415 and D435. These cameras bring the most advanced depth perception capability to your prototype development. The cameras integrate the latest Intel® RealSense™ Vision Processor D4 to handle the complex depth algorithm, the optical module with the high resolution image sensors, Infrared projector and the RGB sensor for color data.

RealSense™ Depth Modules come in different flavors as well: D400, D410, D415, D420 and D430. Please refer to the D400 family datasheet for the detailed description of these modules and their features. They are fully calibrated, turnkey Depth solutions optimized for cost and usages, and ready for system integration and development

Figure 2-1 D400 family cameras



Figure 2-2 D400 family Depth Modules



Intel RealSense D400 series depth cameras produce high quality stereo depth output useful for a variety of market segments including mobile devices, robotics, and volumetric scanning. The quality of depth images produced by the cameras is clearly important. A simple methodology for basic depth image quality evaluation is described above. Intel provides a Depth Quality Tool for easy assessment of the quality of the D400 cameras. The Depth Quality Tool provides a quick and easy way to generate the most relevant depth quality metrics and allows for advanced configuration of the camera so that comparing different configurations is straightforward.

2.2 Depth Quality Tool Introduction






Intel provides an open source real-time tool as part of the Intel® RealSense SDK 2.0 that can be used to compute most of the depth metrics described above and indicate the basic depth image quality of any RealSense™ D400 camera.

Download the Depth Quality Tool for Intel® RealSense™ D400 family of depth cameras from Github.

<https://github.com/IntelRealSense/librealsense/releases/latest>

For Windows installations an executable is available for download

Figure 2-2 Depth Quality Tool download image

Downloads	
 Depth.Quality.Tool.exe	8.35 MB
 Intel.RealSense.SDK.exe	83.5 MB
 Intel.RealSense.Viewer.exe	8.29 MB
 Source code (zip)	
 Source code (tar.gz)	

Linux installation instructions are also available on the Github release page link below:

https://github.com/IntelRealSense/librealsense/blob/development/doc/distribution_linux.md

2.3 Target Considerations

The D400 series camera is used in many scenarios and many markets. To simulate complex scenes the provided target has visual texture. A few target sizes are provided for your convenience. It is recommended to print the target at actual size and not do a "fit to page" printing. It is suggested that the user start with the provided and recommended target for initial testing before moving on to other target designs.

2.4 Setup

2.4.1 Target Distance

Target should be placed between 50 cm and 2 meters away from the camera. A longer distance is possible, but finding large enough targets and keeping them flat will be a challenge. The target should be parallel to the camera imaging plane. The Depth Quality Tool provides real time feedback on the angle of the target for proper alignment.

2.4.2 Lighting

D400 stereo depth when using the provided target will perform well in a wide range of lighting conditions. Typical office or lab lighting is reasonable for testing. The Depth Quality Tool defaults to show a 3D point cloud with visual overlay which makes it easy to determine if the light is good for testing. If the user can see the target in the tool then the lighting is likely good enough.

2.5 Depth Quality Tool Operation

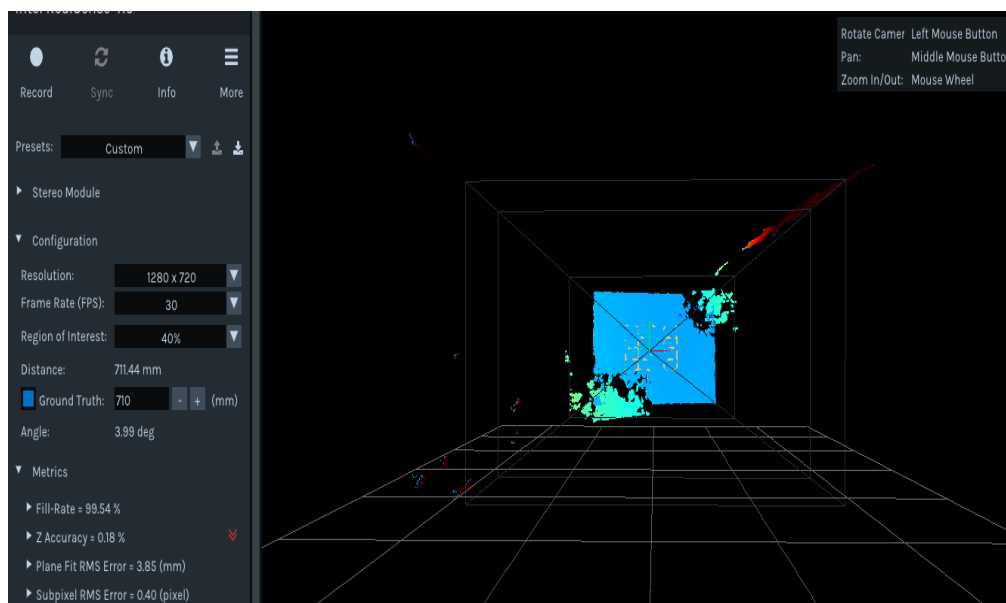
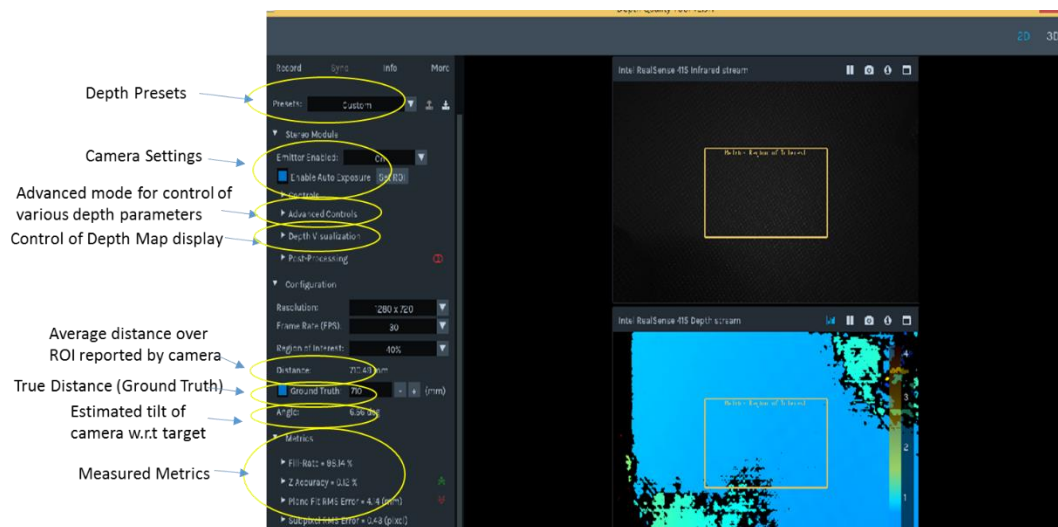
In a windows environment, simply double click on the downloaded Depth.Quality.Tool.exe program to start the Depth Quality Tool.

In Linux, simply run rs-depth-quality to start the Depth Quality Tool.

Figure 2-5 Depth Quality Tool Icon



Figure 2-6 Depth Quality Tool Panel Description



The Depth Quality tool UI consists of a left panel that contains most of the controls, settings, and quality metric values and the right panel displays the real time depth map (top) or 3D point cloud (bottom).

The tool will begin streaming automatically. The yellow box in the center of the frame shows the region of interest used for the test metrics. As the distance grows between target and camera it is difficult to find a target large enough to cover the entire field of

view. The user can select a ROI that covers the target for the appropriate target size and distance. For example, the same target that works at 1 meter with 80% ROI might only cover 40% at 2 meters.

The tool provides feedback to the user if the angle of the imaging plane of the camera is not parallel to the target. For best results the imaging plane should be at most 5 degrees out of parallel from the target plan. Metrics are provided real-time in the bottom left of the window.

It is recommended to start with default settings when testing the D400 camera initially. The user is free to adjust camera settings and get real-time feedback on performance changes. Starting with default settings gives a good baseline of performance to evaluate and build on. New settings can be loaded from the Preset pull down menu.

In cases where a camera does not meet one or more of the performance criteria stated in the D400 Datasheet, it is suggested that the test be repeated after the recommended test conditions and camera settings are confirmed. If any of the performance criteria are confirmed to not meet the recommended values, Intel offers a few tools that can be used to re-calibrate the unit – Dynamic calibration (field) and OEM calibration (factory/engg. environment)

2.5.1 Reporting

The Depth Quality Tool provides an easy reporting mechanism for saving all relevant data associated with the camera. Included in the save are the images from the right and left cameras, the depth frame, all of the settings, and the firmware version. This data can be used for comparisons between multiple settings or could be sent to Intel support for troubleshooting purposes.

2.6 Subjective Testing Considerations

The first part of this document focuses on quantitative testing methodology, the second part is focused on the RealSense Depth Quality Tool which provides an easy method of obtaining well understood quantitative results using a flat target at no more than a few meters away. This type of testing is valuable and necessary because it can be controlled and repeated and provides quantitative metrics that can be used to objectively evaluate camera image quality. It is however, useful and necessary to also test depth cameras in real-world environments that do not permit such quantitative testing. This section of the document will review some important considerations when testing depth cameras and how to interpret the results.

2.6.1 Lighting

Depth cameras use sensors that collect light and then interpret that data to create a depth map. Some sensors collect light emitted from the depth camera itself, e.g., structured light (SL) and time of flight (ToF), and some sensors collect the ambient light from the scene, e.g., passive stereo cameras. The D400 series is a hybrid of these using ambient light and a projected pattern when necessary.

If the depth camera will be used in various lighting conditions then it should be tested in a variety of lighting conditions. Different technologies will behave differently in certain lighting conditions and these tradeoffs must be well understood so that a single controlled lab test does not bias results.

Keep in mind that lighting changes will affect the sensor and if a filter is used to block certain wavelengths of light from reaching the sensor then that should be taken into account. For example, it may not be necessary to test a ToF camera in total darkness because it likely has a filter which blocks all visible light anyway. Conducting a test in an environment that has no ambient light in the range of 830-870nm (assuming the system is designed to operate with 850nm near IR) is equivalent to testing in a completely dark room.

It is recommended to test depth cameras in at least three conditions - dark, home or office lighting, and sunlight. Dark, or nearly dark, tests are mostly required for passive systems. Sunlight tests are mostly for ToF and SL systems which will need to extract their projected light from the background IR of the sun. Mixed lighting tests may also be required depending on the usage model. For example, if a depth camera is to be used inside a home during the day then some amount of sunlight will be present.

It is possible to use the DQT in these different lighting conditions if the testing environment allows for it.

2.6.2 Materials

As mentioned earlier in this document the traditional test target for ToF and SL is a flat white surface of about 85% reflectivity and it is recommended to use a visually textured target with stereo camera systems like the D400. Other materials of various reflectivity should be tested based on how the camera will be used. For an autonomous vacuum cleaner usage it would be prudent to test different floor materials such as light and dark tile, wood, carpet, and linoleum. For a gesture control system in a car it is necessary to test different glove material including leather. For body scanning usages it is necessary to test materials that would be worn by the user and different colors/patterns of that material.

In general, IR-based systems such as ToF and SL will have difficulty with materials that do not reflect IR well. These include leather, dark clothing, and dark carpets. In general, stereo systems will have difficulty with materials that have no visual texture like solid color tile floors. The D400 employs a projector that adds visual texture to these types of materials.

All visual based depth cameras will have difficulty with reflective surfaces to varying degrees.

It is possible to use the DQT on different materials if care is taken to use a target that is flat. If something like a carpet is used then there should be an expectation that the RMS error will be higher since the carpet is made up of material that is varying in height.

2.6.3 Distance

The RealSense Depth Quality Tool does not limit the maximum distance tested, but finding a target large enough to fill an adequately sized region of interest as distance increases beyond three meters is difficult. Long distance testing likely requires subjective analysis. The quality of depth required at distances beyond 5 meters is defined by the usage. For example, a room scanning device designed to remain in the center of a large space and rotate while capturing walls, ceiling, and objects in detail will require accurate depth at long distance. In contrast, a robot in motion can use depth at longer distances for path planning without requiring accurate depth at those distances since a path can be planned to avoid a tree whether the tree is 9 meters away or 10 meters away.

Triangulation-based systems like SL and Stereo have reduced depth accuracy as distance increases. IR-based systems like ToF and SL are limited by the sensitivity of the sensor used and the strength of the projector. The projected light must be able to reflect off of the object at distance and return to the sensor or no depth data can be generated. Note that SL systems are limited in distance both by the power of the projector and the size of the baseline.

2.6.4 Shape

Quantitative testing of depth is normally done using flat targets primarily because establishing ground truth and interpretation of the data are relatively simple. Using scenes with more complex shapes is necessary, but difficult to quantify. Qualitative analysis likely includes looking at the point cloud for edge fidelity, flat or round surfaces, and proper angles. Again, it is important to understand the use case in order to focus testing efforts, but some recommended objects and shapes include spheres, outer and inner corners, stepped surfaces, and small or thin objects.

