Quick Start Guide
SBOU157 – September 2015

# Voxel Viewer

The purpose of this user manual is to familiarize Voxel Viewer users with the software, including the GUI layout, and navigation around the menus, windows, and controls. This document also covers the various calibration procedures required for accurate measurements. Chapter 1 provides an introduction to the 3D time-of-flight (ToF) technology and provides an overview of the contents. Chapter 2 covers the user interface details, including the various display and control elements. Chapter 3 covers the basic operation of the Voxel Viewer, including setting of parameters, import and export of data. Chapter 4 covers the calibration that includes lens calibration, modulation frequency calibration, crosstalk and nonlinearity calibration, common phase calibration, and finally, pixel-wise calibration. The last chapter, Chapter 5, covers the time-of-flight controller (TFC) programming.

**Contents**

**List of Figures**

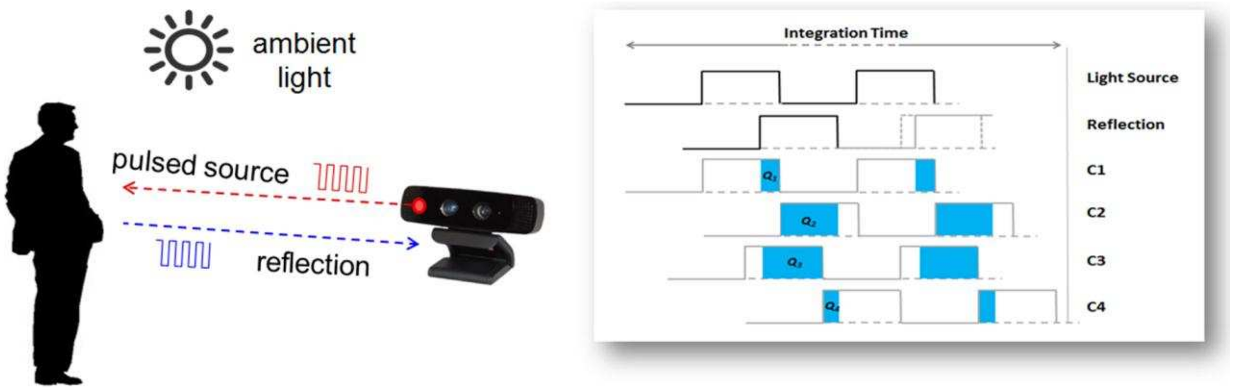
# 1 Introduction

## 1.1 *TI 3D Depth Sensors*

Texas Instruments offers a family of 3D Depth Sensor chipsets. Each chipset includes an imaging component and a digital component, implemented as 1-chip, 2-chip, or 3-chip solutions. The first generation chipset, codename “Haddock”, is a 3-chip solution (OPT8140+VSP5324+OPT9220). The second generation chipset, codename “TinTin”, is a 2-chip solution (OPT8241+OPT9221); both have a 320x240 pixel resolution. Future generations will include 1-chip or 2-chip solutions in different pixel resolutions.

## 1.2 *Operating Principles*

All of these chipsets measure depth based on ToF principle where the scene is illuminated with modulated light at 850-nm wavelength. The phase difference between the emitted light and the light reflecting off the scene is measured, per pixel, and converted to depth; then the depth image is converted to point clouds, where each pixel is transformed into 3D coordinates with the origin at the optical center of the 3D ToF imager.



$$\phi = \arctan\left(\frac{Q_3 - Q_4}{Q_1 - Q_2}\right) \qquad d = \frac{c}{2f} \cdot \frac{\phi}{2\pi}$$

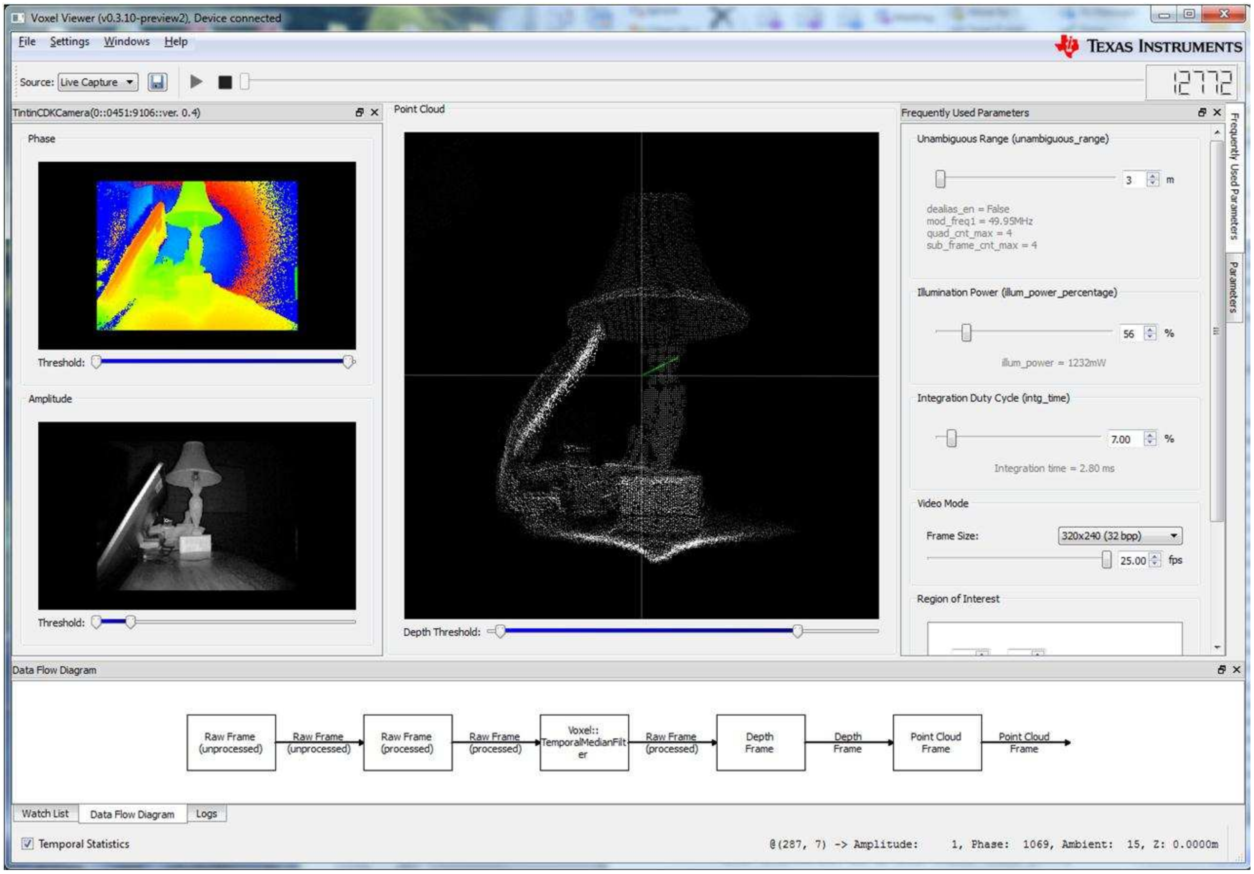
**Figure 1. 3D Time-of-Flight Sensor Operating Principle**

## 1.3 *Voxel Viewer and Voxel SDK*

To facilitate rapid time-to-market, TI offers an open-source software development kit (SDK), called Voxel SDK, from which one can develop custom 3D-ToF applications quickly. Furthermore, Voxel SDK contains sample applications that show how use the available APIs. The Voxel Viewer is a graphical application developed by TI that uses the Voxel SDK. The Voxel SDK enables users to evaluate the TI 3D ToF solutions without having to write any code.

## 2 User Interface

The user interface is composed of the top-level menu bar, play control, main viewport, and left, right, and bottom panes where dialog boxes can dock. Figure 2 shows the Voxel Viewer top-level layout.



Source: Live Capture

Unambiguous Range (unambiguous_range): 3 m

dealias_en = False
mod_freq1 = 49.95MHz
quad_cnt_max = 4
sub_frame_cnt_max = 4

Illumination Power (illum_power_percentage): 56 %

illum_power = 1232mW

Integration Duty Cycle (intg_time): 7.00 %

Integration time = 2.80 ms

Video Mode

Frame Size: 320x240 (32 bpp)

25.00 fps

Region of Interest

☑ Temporal Statistics

```
@(287, 7) -> Amplitude:     1, Phase:  1069, Ambient:  15, Z: 0.0000m
```

**Figure 2. Voxel Viewer Top-Level**

### 2.1 Menu Bar

The menu bar is found at the top of the Voxel Viewer application window; it contains four major menu items: **File**, **Setting**, **Windows**, and **Help**.

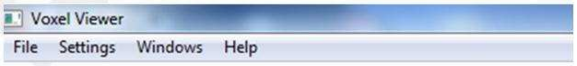


**Figure 3. Voxel Viewer Menu Options**

### 2.1.1 File

Selecting **File** will bring up a drop-down selection box for connecting/disconnecting depth camera, opening saved data stream, and opening the programmer to download new firmware to the CDK. Application can exit by selecting *Exit*. Note that each menu item has an associated fast keyboard access to the command.

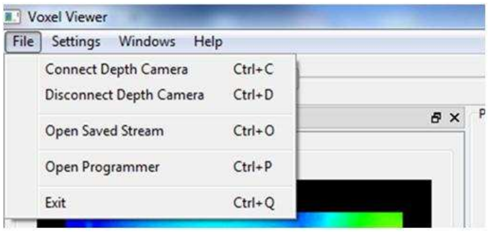


**Figure 4. File Menu Drop-Down Box**

### 2.1.2 Settings

Selecting **Settings** brings up another drop-down box with two more options: **Camera Profiles** and **Statistics**.

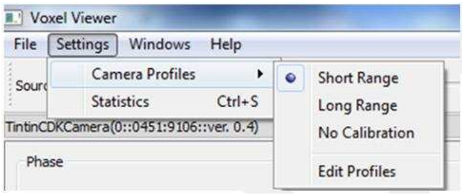


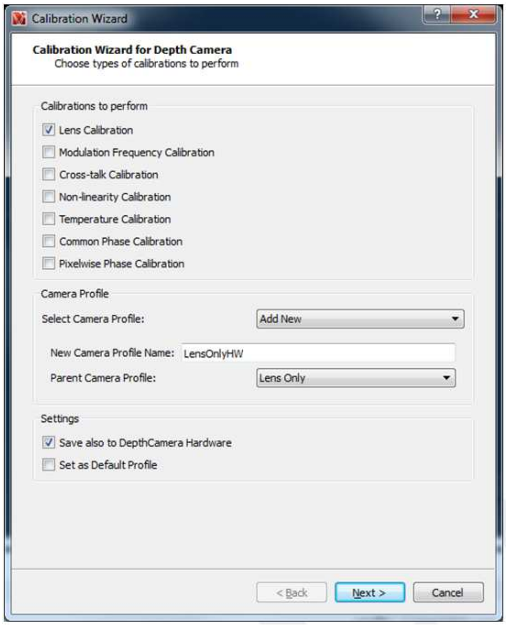
**Figure 5. Settings Menu Drop-Down Box**

#### *2.1.2.1 Camera Profiles*

**Camera Profiles** further lets users select from operating the ToF target in **Short Range**, **Long Range**, and **No Calibration** modes. The **Long Range** mode and **Short Range** mode differ in that the **Long Range** mode deploys the chipset's de-aliasing feature to extend the range beyond the unambiguous distance. Please read *Time-of-Flight Camera: An Introduction* (SLOA190) for more information on de-aliasing and unambiguous distance. The purpose of the **No Calibration** selection is to enable one to compare and contrast the effect of calibration. New profiles can be created through the **Edit Profiles** option.

### 2.1.2.2 Edit Profiles

**Edit Profiles** brings up the following window, which has two sections. The top section gives users the ability to select the types of calibration to perform; the bottom section allows the user to name and save the profile. Calibration is discussed in detail in Section 4.1–Section 2.6.



Calibration Wizard

**Calibration Wizard for Depth Camera**
Choose types of calibrations to perform

Calibrations to perform

☑ Lens Calibration
□ Modulation Frequency Calibration
□ Cross-talk Calibration
□ Non-linearity Calibration
□ Temperature Calibration
□ Common Phase Calibration
□ Pixelwise Phase Calibration

Camera Profile

Select Camera Profile: Add New

New Camera Profile Name: LensOnlyHW

Parent Camera Profile: Lens Only

Settings

☑ Save also to DepthCamera Hardware
□ Set as Default Profile

Figure 6. Edit Profile Dialog Box

Each profile under **Settings** → **Profiles** corresponds to a set of calibration parameters. A profile is defined by the following set of parameters:

- unambiguous_range
- frame_rate
- sub_frame_cnt_max
- quad_cnt_max
- intg_time
- illum_power
- mix_volt
- fsc_enable
- delay_fb_corr_en
- delay_fb_dc_corr_en

These parameters are adjusted to adapt the 3D-ToF camera to different types of applications.

### 2.1.2.3 Statistics

**Statistics** shows a collection of statistics in the **Watch List** window. If the **Temporal Statistics** checkbox at the bottom left corner of the main window is selected, temporal statistics are displayed in the **Watch List** window along with the *spatial* statistics.

### 2.1.3 Windows

The **Windows** menu option lets users chose which windows are displayed. There are several options, and each are discussed in the latter part of this document.

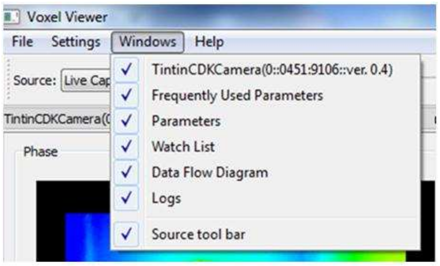


**Figure 7. Windows Menu Drop-Down Box**

### 2.1.4 Help

The **Help** menu provides helpful information on how to use Voxel Viewer, including displaying the license information and the software version number.

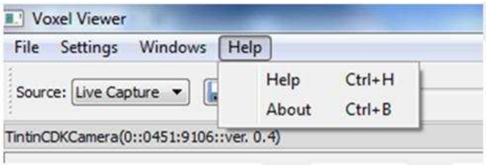


**Figure 8. Help Menu Drop-Down Box**

## 2.2 Source Bar

The **Source** bar is beneath the top-level menu. It allows the user to select the input source, which can be from a live stream or a recorded stream. The Play and Stop buttons start and stop the streaming. The floppy disc icon starts a recording, which once stopped, will save the stream to a file. The progress bar shows playback progress, and at the end of the progress bar is the frame counter that shows the cumulative number of frames captured.

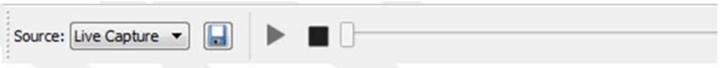


**Figure 9. Source Bar**

## 2.3 Main Viewport

The main viewport is the largest display window, and is best used for visualizing 3D point clouds, though other data can be displayed in this viewport as well. To choose the data source, simply right-click on the frame of the main viewport window to bring up a selection box. The different data sources are described in Section 3.3.

At the bottom of the Main Viewport is a double-ended slider. The adjustment sliders are used to clip the point clouds based on the *z*-value of each pixel—the left slider sets the lower-limit, and the right slider sets the upper-limit. Only those points whose *z*-values are between the limits are displayed.

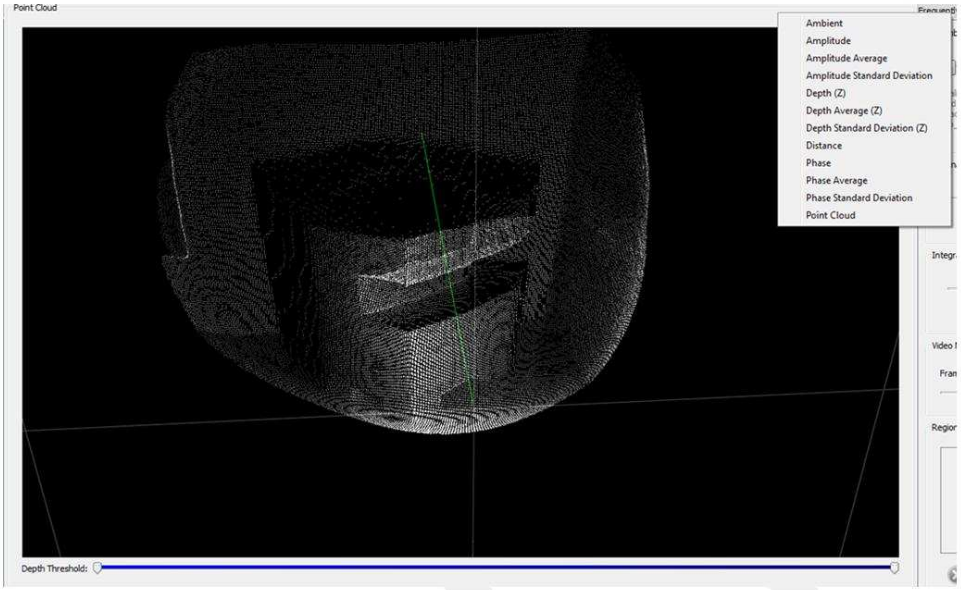


**Figure 10. Main Viewport**

## 2.4 Left Pane

By default, the left pane is where the side viewports are docked.

### 2.4.1 Side Viewports

By default the side viewports show the distance map at the top, and amplitude map at the bottom. But each of these two viewports can also display other data maps by right-clicking the viewport's frame area to bring up the selection box. Histogram display can also be selected. Figure 11 shows the side viewports with and without the histograms.

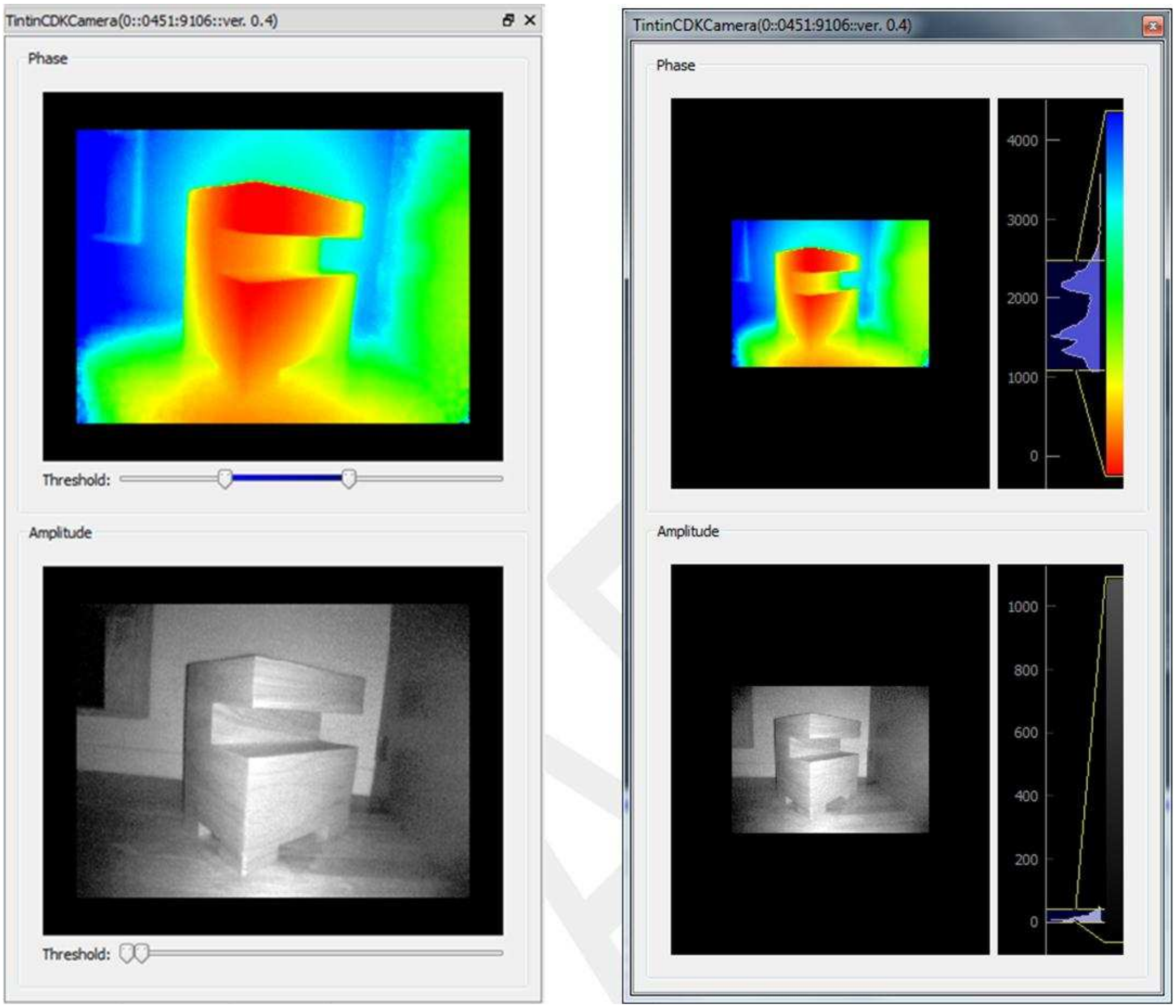
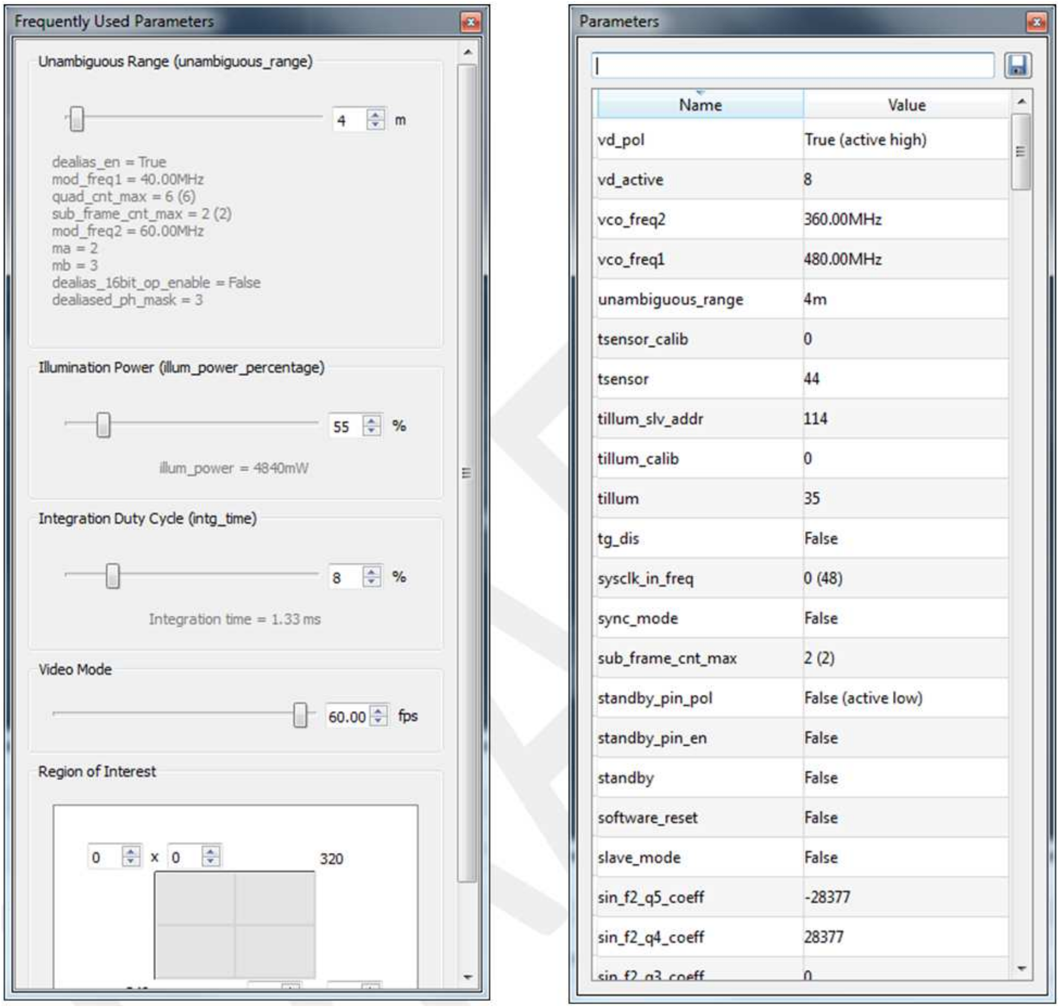


**Figure 11. Side Viewports (Left); With Histogram Enabled (Right)**

## 2.5 Right Pane

The right pane, by default, holds the **Frequently Used Parameters** window and the **Parameter List** window. These two windows provide access to the ToF camera hardware registers.



Frequently Used Parameters

Unambiguous Range (unambiguous_range): 4 m

dealias_en = True
mod_freq1 = 40.00MHz
quad_cnt_max = 6 (6)
sub_frame_cnt_max = 2 (2)
mod_freq2 = 60.00MHz
ma = 2
mb = 3
dealias_16bit_op_enable = False
dealiased_ph_mask = 3

Illumination Power (illum_power_percentage): 55 %

illum_power = 4840mW

Integration Duty Cycle (intg_time): 8 %

Integration time = 1.33 ms

Video Mode: 60.00 fps

Region of Interest: 0 x 0 320

Parameters

| Name | Value |
|---|---|
| vd_pol | True (active high) |
| vd_active | 8 |
| vco_freq2 | 360.00MHz |
| vco_freq1 | 480.00MHz |
| unambiguous_range | 4m |
| tsensor_calib | 0 |
| tsensor | 44 |
| tillum_slv_addr | 114 |
| tillum_calib | 0 |
| tillum | 35 |
| tg_dis | False |
| sysclk_in_freq | 0 (48) |
| sync_mode | False |
| sub_frame_cnt_max | 2 (2) |
| standby_pin_pol | False (active low) |
| standby_pin_en | False |
| standby | False |
| software_reset | False |
| slave_mode | False |
| sin_f2_q5_coeff | -28377 |
| sin_f2_q4_coeff | 28377 |
| sin_f2_q3_coeff | 0 |

**Figure 12. Frequently Used Parameters Dialog Box (Left), and Parameter List Dialog Box (Right)**

### 2.5.1 Frequently Used Parameters Window

The **Frequently Used Parameters** window translates simple user interface actions into the more complex underlying register access sequences, such as ***unambiguous range***, illumination power, integration duty cycle, video mode, and region of interest (ROI). The unambiguous range defines the maximum range one wants to measure, and will deploy anti-aliasing when necessary to achieve the desired range. The text beneath the slider shows the register settings implementing the range specification. The ***illumination power*** controls the illumination emission power, specified as a percentage of the maximum power permitted by the hardware. The ***integration duty cycle*** specifies the percentage of frame time where the sensor is exposed to reflected illumination. The ***video mode*** enables the user to subsample the full image at different resolution; and select an allowed frame rate for the given resolution. The ***region-of-interest (ROI)*** specifies a rectangle area to stream, excluding other parts of the image.

### 2.5.2 Parameter List Window

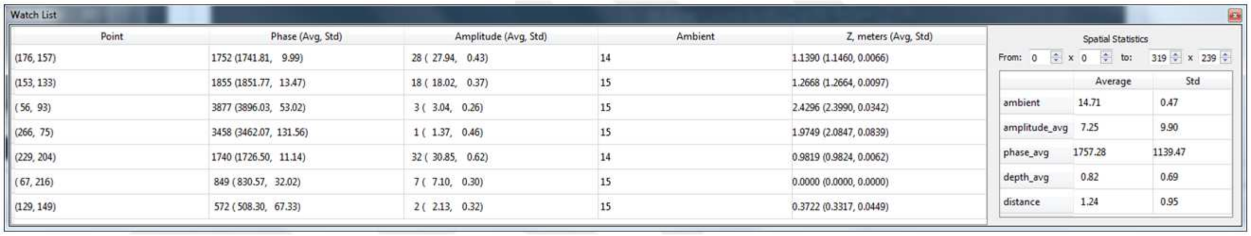
The parameter list box gives the low-level register access. Usually this window is unused. Advanced users can adjust these parameters for optimal operation. Users wishing to use these registers are strongly encouraged to first read the *Introduction to Time of Flight Camera – An Introduction, and the Introduction to the ToF System Design* (SLOA190), and the Time-of-Flight Controller (TFC) data sheet of your specific camera, such as OPT9221. The parameter values can be saved to a *.csv file by clicking on the “floppy disc” icon at the upper right-hand corner.

## 2.6 Bottom Pane

The bottom pane, by default, docks windows for Watch List and Statistics, Data Flow Diagram, and message log.

### 2.6.1 Watch List and Statistics Window

The Watch List and Statistics can be used to add points from the Point Cloud display to watch. To add a new watch point, simply select the point you want to watch in a side viewport by left-clicking while holding down the **Shift** key. The statistics box on the right shows the various spatial statistics of the image amplitude, phase, and ambient.



Watch List

| Point | Phase (Avg, Std) | Amplitude (Avg, Std) | Ambient | Z, meters (Avg, Std) |
|---|---|---|---|---|
| (176, 157) | 1752 (1741.81, 9.99) | 28 ( 27.94, 0.43) | 14 | 1.1390 (1.1460, 0.0066) |
| (153, 133) | 1855 (1851.77, 13.47) | 18 ( 18.02, 0.37) | 15 | 1.2668 (1.2664, 0.0097) |
| ( 56, 93) | 3877 (3896.03, 53.02) | 3 ( 3.04, 0.26) | 15 | 2.4296 (2.3990, 0.0342) |
| (266, 75) | 3458 (3462.07, 131.56) | 1 ( 1.37, 0.46) | 15 | 1.9749 (2.0847, 0.0839) |
| (229, 204) | 1740 (1726.50, 11.14) | 32 ( 30.85, 0.62) | 14 | 0.9819 (0.9824, 0.0062) |
| ( 67, 216) | 849 ( 830.57, 32.02) | 7 ( 7.10, 0.30) | 15 | 0.0000 (0.0000, 0.0000) |
| (129, 149) | 572 ( 508.30, 67.33) | 2 ( 2.13, 0.32) | 15 | 0.3722 (0.3317, 0.0449) |

Spatial Statistics

From: 0 x 0 to: 319 x 239

| | Average | Std |
|---|---|---|
| ambient | 14.71 | 0.47 |
| amplitude_avg | 7.25 | 9.90 |
| phase_avg | 1757.28 | 1139.47 |
| depth_avg | 0.82 | 0.69 |
| distance | 1.24 | 0.95 |

**Figure 13. Watch List and Statistics**

### 2.6.2 Data Flow Diagram Window

The **Data Flow Diagram** is used to display the data pipeline from the raw phase data to the final point cloud display. It is mostly useful to show the inserted filter components.

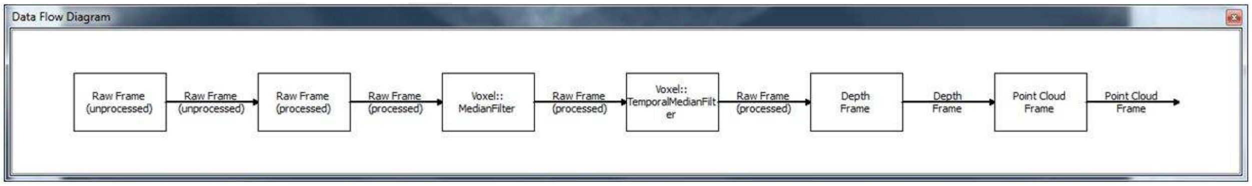


**Figure 14. Data Flow Diagram**

Software filters can be added between the **Raw Frame (processed)** and the **Point Cloud Frame**. To add a filter, right-click on any component blocks and choose to insert the filter before or after the component. A filter can also move to the right or to the left, or altogether be removed, using the drop-down box.

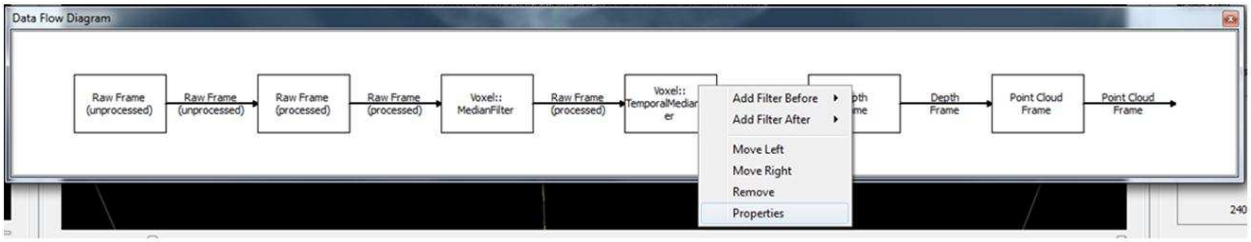
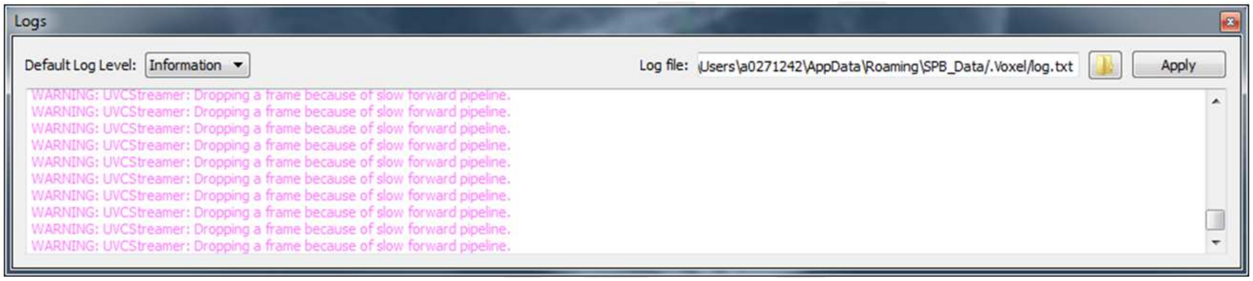


**Figure 15. Filter Drop-Down Menu**

### 2.6.3 Logs Window

During operation, Voxel Viewer may display warning or error messages in the Logs dialog box to inform users of its internal state. Different types of information can be logged: critical, error, warnings, information and debug. The log can be exported to a file for debug purposes.



Default Log Level: Information

Log file: \Users\a0271242\AppData\Roaming\SPB_Data/.Voxel/log.txt

```
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
WARNING: UVCStreamer: Dropping a frame because of slow forward pipeline.
```

**Figure 16. Logs Window**

# 3 Basic Operations

## 3.1 Connecting Devices

Voxel Viewer is capable of connecting to multiple streams, but can only render one stream at a time. To connect a device to Voxel Viewer, simply plug in the ToF camera into the USB port. Only devices supported by the Voxel SDK will be detected and enumerated.

A compatible ToF camera can be connected after the Voxel Viewer is launched. If the Voxel Viewer is launched before the device is connected, the Voxel Viewer will display an inactive screen like the following image. From there, one can still connect a camera, just click the **Refresh** button to tell Voxel Viewer to rescan for attached ToF cameras.

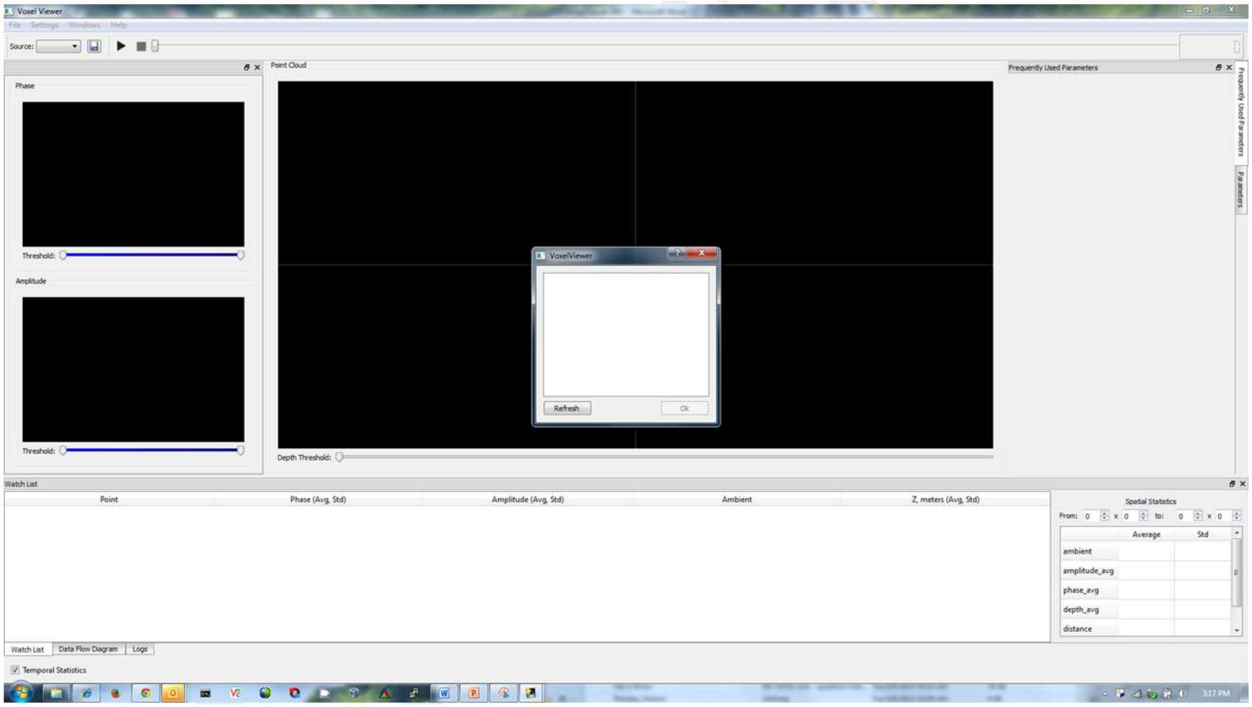


**Figure 17. Voxel Viewer Launched Without Connecting a Camera**

## 3.2 Adjusting the Settings

After Voxel Viewer is launched, by default the left viewports will show color-coded distance map at the top, and amplitude map at the bottom. The main viewport will show point clouds. After launch, to achieve the best image quality, follow these general steps:

1. Set the desired unambiguity range.
2. Set the desire video mode and frame rate.
3. Point the camera at the target or the scene; with the integration period set to 8% to 15%, adjust the illumination power so the scene can be seen reasonably well.
4. Insert a **Bilateral filter** or **Median filter** block after the **Raw Frame** (processed) block.
5. Insert a **Temporal Median Filter** after the **Bilateral Filter**.
6. Tune the **Bilateral Filter** and **Temporal Median Filter** for best results.

## *3.3 Visualizing the Data*

The Voxel Viewer supports display of **amplitude**, **phase**, **ambient**, **depth**, **point clouds**, and various derived data from amplitude and phase.

### 3.3.1 Ambient

The ambient value of a pixel represents the sensed values *not* from the modulated illumination source. This value may include systematic offsets, such as electrical propagation delay, pixel integration property variation, and other ambient light sources in the near-infrared (NIR) band. When displayed, one will see a faint image of the scene plus vertical stripes caused by different electrical propagation delay from different pixel columns.

### 3.3.2 Amplitude

The amplitude map shows each pixel’s return amplitude in a 12-bit value (0–4095). The amplitude map can be treated as a grayscale image, and processed using popular image processing algorithms, such as blob and edge detection. In general, the higher the amplitude, the more accurate a pixel’s phase measurement; therefore, the amplitude map is also called *confidence map* by some, and it can be used to weigh successive phase measurements in an average filter.

### 3.3.3 Depth

The depth map is a map of the *z*-dimension of pixels in the point cloud. The point clouds conversion is first calculated, then the *z*-dimension of each pixel is extracted and arranged as a map. The depth map is particularly useful for range measurement or robotic navigation purposes. The values are in millimeters.

### 3.3.4 Distance

The distance map is a map of the shortest distance (or the Euclidean distance) from the camera origin to the point in the scene, through the pixel where the point is imaged. The distance map is directly proportional to the phase map outputted by the time-of-flight controller (TFC). The values are in millimeters.

### 3.3.5 Phase

The phase map is a measurement of each pixel’s relative phase between the emitted light and the reflected light. The phase is nearly proportionally with the actual distance. The difference between raw phase and processed phase is the phase offset. The phase offset may be different for each pixel, and may be a function of temperature and received amplitude. Correcting for the phase offset is the main focus of the calibration process.

### 3.3.6 Point Clouds

The point clouds data are derived from the depth map, where each pixel ($u$, $v$, $d$), in the depth map is transformed a Cartesian coordinate ($x$, $y$, $z$). The resulting points can be rendered in a 3D virtual space for visualization. The point clouds are usually used in 3D visualization and scanning and in auto-dimensioning applications where precise measurement of each dimensions is important.

### 3.3.7 Histogram

Histograms can be generated for the side viewports. It bins the image values to show the value distribution over the pixel range. Histograms are great for automatic gain control or adjustments based on distribution of pixel values.

## 3.4 De-Noising

Noise can be reduced by both temporal filters and spatial filters. Temporal filters suppress high temporal changes, and spatial filters suppress high spatial changes.

### 3.4.1 Temporal Filters

Voxel Viewer includes two temporal filters: IIR filters and median filter. Temporal filters are good at removing temporal noise, but often at the expense of introducing motion blur.

#### 3.4.1.1 IIR Filter

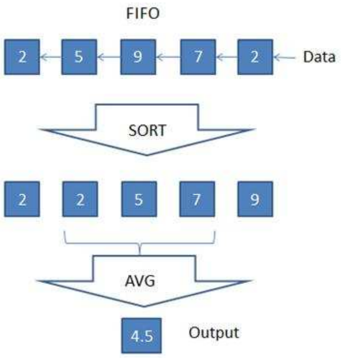
The infinite-impulse-response filter (IIR) is a simple first order filter of the form:

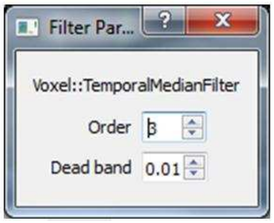
$$X_{new} = (1 - \alpha)x_{old} + \alpha x_{in} \tag{1}$$

Where $\alpha \in (0..1)$. The filter responds to change slower (stronger filter) for larger α, but this will also lead to greater motion blur. Therefore, IIR filter is better suited for static scenes, such as in 3D scanning applications.

#### 3.4.1.2 Median Filter

The temporal median filter samples by feeding data into a FIFO, sorting the buffer values, and taking the median value as the output. This effectively removes the extreme values, or noise, if the noise is characterized by intermittent spikes of extreme values. In ToF cameras, the “flying-pixels” effect can be reduced by this filter. The default dead band threshold is 0.05 (5%); that means if the new filtered value differs from the current value by less than 5%, the pixel does not change. This has a stabilizing effect on the output image.





Voxel::TemporalMedianFilter

Order 3

Dead band 0.01

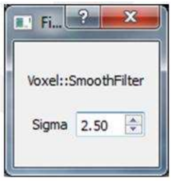
**Figure 18. Temporal Median Filter Algorithm (Left) and Properties (Right)**

### 3.4.2 Spatial Filters

Spatial filters attempt to reduce the variations within a 2D image. These filters generally do not introduce temporal delays, as long as the CPU can keep up with the calculation.

#### 3.4.2.1 Smooth Filter

The smooth filter is a spatial filter that applies a 2D Gaussian function to compute weighted average about the square kernel centered about a given pixel. The effect of the Gaussian function is smoothing of high-frequency elements (for example, sharp spatial changes), such as speckle noise. It is equally important to note that the actual scene may contain high-frequency structural elements, such as edges, will be smoothed by this filter also. The property, ***sigma***, specifies how much smoothing is applied—the larger the value, the more smooth (and blurry). Generally the smooth filter is applied to the phase map.



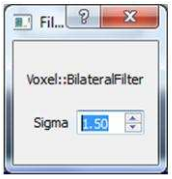
Voxel::SmoothFilter

Sigma 2.50

**Figure 19. Smooth Filter Property**

#### 3.4.2.2 Bilateral Filter

The bilateral filter is an enhanced form of smooth filter that retains the high-frequency structure in the phase map by using the amplitude map as a reference. This filter assumes neighboring pixels with similar amplitudes as the center pixel have similar depth; therefore, the center pixel's filtered value is an average of its neighboring pixels weighed amplitudes similarity. The bilateral filter has ***sigma*** as a property that defines the degree of smoothing. Generally the bilateral filter is applied to the phase map.



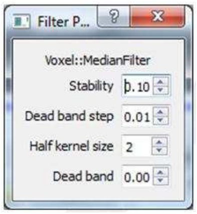
Voxel::BilateralFilter

Sigma 1.50

**Figure 20. Bilateral Filter Property**

#### 3.4.2.3 Median Filters

The spatial median filter, not to be confused with the temporal median filter, is also designed to remove spatial noise; but it is done by ordering the pixels in a square sub-image (kernel) and then taking the median value as the center pixel's filtered output. Since the median filter uses the measured pixel value rather than interpolated pixel value, the filtered image is generally a better reproduction of the actual scene. The spatial median filter has several properties. The most important of all is the half-kernel size, which defines a kernel of size of 2n + 1. For the example shown in Figure 21, the kernel size is 5. The ***dead band step*** and ***stability*** should be generally untouched at 0.01 and 0.10, respectively.



Voxel::MedianFilter

Stability 0.10

Dead band step 0.01

Half kernel size 2

Dead band 0.00

**Figure 21. Spatial Median Filter Properties**

### 3.4.3 Recommended Starting Point

A recommended starting point for de-noising is to cascade a median filter or a bilateral filter, followed by temporal median filter. From there the filters can be fine-tuned to optimize for desired performance. The data flow diagram in Figure 14 is one of the recommended starting points.

# 4 Calibration

Voxel Viewer supports several types of calibration: lens calibrations, modulation frequency calibration, crosstalk calibration, nonlinearity calibration, temperature calibration, common phase calibration and pixel-wise calibration. In this section, the different types of calibration are first described; then the last section provides a summary of how these individual calibrations come together for setup of different operating profiles. Calibration is a complex process, an advanced operation that only needs to be performed occasionally. Readers are advised to read through this entire section before attempting calibration.

## 4.1 Lens Calibration

Lens calibration aims at overcoming optical distortions introduced by the lens. In absence of any optical distortions, each pixel ($u_l$, $v_i$) can be converted to the equivalent world coordinates ($x_i$, $y_i$, $z_i$) by applying the pin-hole camera model shown in Figure 22. The goal of lens calibration is to transform the distorted pixel coordinates to equivalent coordinates that can be used under the pin-hole camera model.

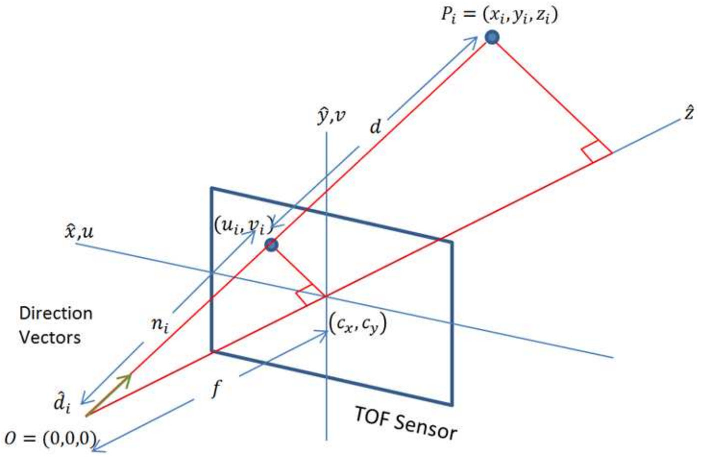
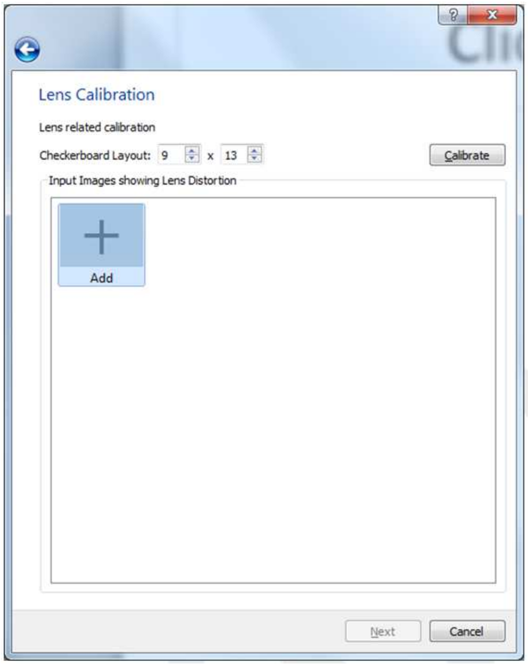


**Figure 22. Pin-Hole Camera Model**

The process extracts a set of parameters commonly known as *intrinsics* that include nine values [$k_1$, $k_2$, $k_3$, $p_1$, $p_2$, $c_x$, $c_y$, $f_x$, $f_y$], where [$k_1$, $k_2$, $k_3$] are the radial distortion coefficients, [$p_1$, $p_2$] are the tangential distortion coefficients, and [$c_x$, $c_y$] are the coordinates of the image center, and [$f_x$, $f_y$] are the *x* and *y* focal distances.

The procedure takes several amplitude images of a checkerboard pattern, similar to the one shown in Figure 24. The checkerboard corners are extracted as data points used in computing the intrinsics through least-square minimization. Users wishing to learn about the theory behind lens calibration are encouraged to review the OpenCV tutorial link in the **Reference** section.

To perform lens calibration, first select **Settings** → **Edit Profiles** to bring up the profile window. Make sure the lens calibration checkbox is selected. Add a new profile by providing a new profile name, then click **Next**, and the lens calibration window as shown in Figure 23 displays.



Lens Calibration

Lens related calibration

Checkerboard Layout: 9 x 13

Input Images showing Lens Distortion

**Figure 23. Lens Calibration Window**

Lens calibration will require a flat, rigid checkerboard pattern like the one shown in Figure 24. Please contact TI for the checkerboard pattern. The checkerboard should be printed on A2 sized paper.

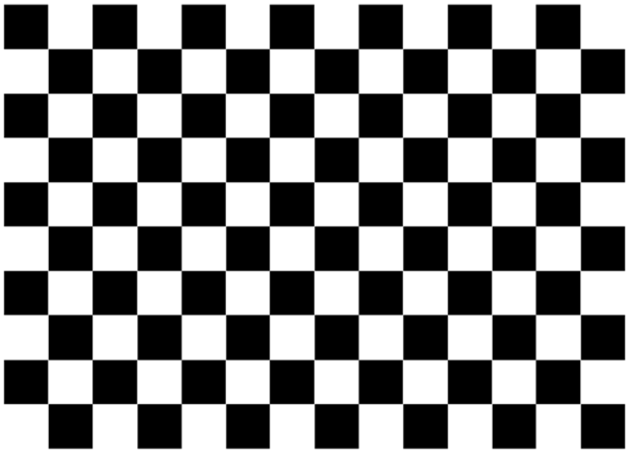
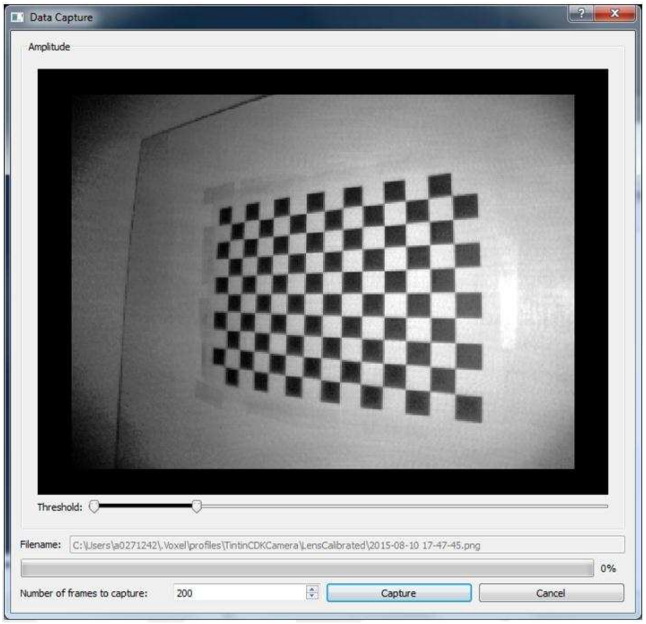


**Figure 24. Checkerboard Pattern for Lens Calibration**

To start calibration, make sure the camera is running then click **Add** to bring up the image capture window shown in Figure 25. Some of the important considerations when performing lens calibration are:

1. Filling the screen as much as possible with the entire checkerboard pattern
2. Presenting the pattern in different poses—normally significant translation is not needed, it is more important to present the pattern at different orientations
3. Keep the checkerboard pattern still while capturing is taking place. Multiple samples (default is 200 frames) will be captured and averaged to remove noise, before the corner points are extracted for calibration.



Filename: C:\Users\a0271242\.Voxel\profiles\TintinCDKCamera\LensCalibrated\2015-08-10 17-47-45.png

Number of frames to capture: 200

Figure 25. Checkerboard Capture Window

One can verify whether a capture is good is by double-clicking on the captured image. Figure 26 shows a good captured image. A good capture will have a zig-zag colored pattern going through all the corner points overlaid on top of the image. If the scan pattern is in entirely in red, or if the image showed no overlaying pattern, then the capture is not good, and the captured image must be discarded and a new capture repeated.

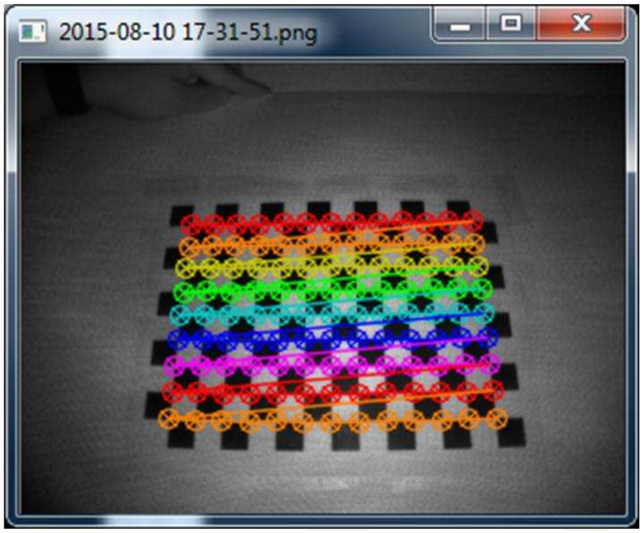
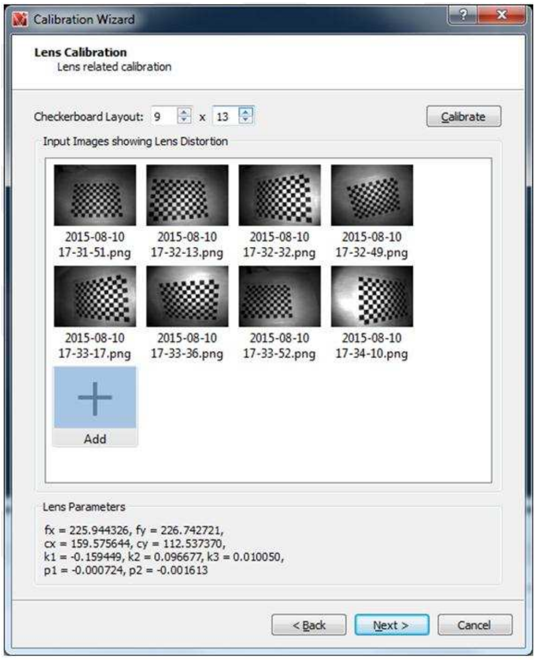


Figure 26. Example of a Good Capture

After capturing at least 8 different views, something equivalent to the following window should result. Now click **Calibrate** to compute the *intrinsic* parameters, which will display at the bottom left-hand-corner. Click **Next** to conclude lens calibration.



Calibration Wizard

Lens Calibration
Lens related calibration

Checkerboard Layout: 9 x 13

Input Images showing Lens Distortion

Lens Parameters

```
fx = 225.944326, fy = 226.742721,
cx = 159.575644, cy = 112.537370,
k1 = -0.159449, k2 = 0.096677, k3 = 0.010050,
p1 = -0.000724, p2 = -0.001613
```

Figure 27. Lens Calibration After 8 Captures

## 4.2 Frequency Calibration

Frequency calibration determines the scale factor necessary to compensate frequency register settings such that the desired frequency is produced. Without frequency calibration, a scale factor of 1 is assumed.

### 4.3 Crosstalk Calibration

Crosstalk calibration overcomes a shift in the phase measurement due to pixel crosstalk, which can lead to *embossing* of borders where there are high amplitude contrasts. This artifact can be mitigated by convolving the phase map with a filter kernel function to null out phase contribution from the surrounding pixels.

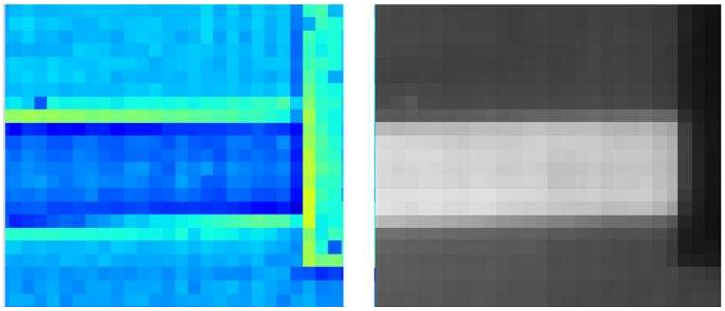
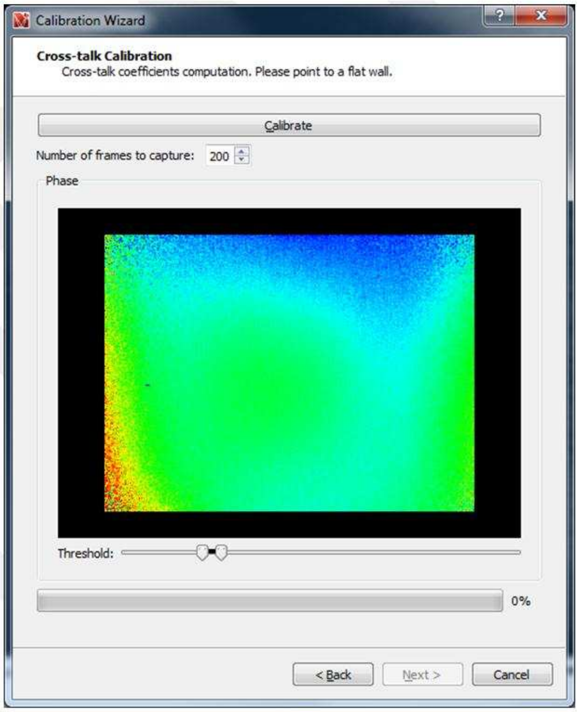


**Figure 28. Phase Shift at the Boundary of High Amplitude Variation has Extra Offset**

To perform crosstalk calibration, place a small reflector at a distance such that the reflector becomes a single pixel in the image. Click **Calibrate** to sample the image for extracting the filter kernel function.



Calibration Wizard

**Cross-talk Calibration**

Cross-talk coefficients computation. Please point to a flat wall.

Number of frames to capture: 200

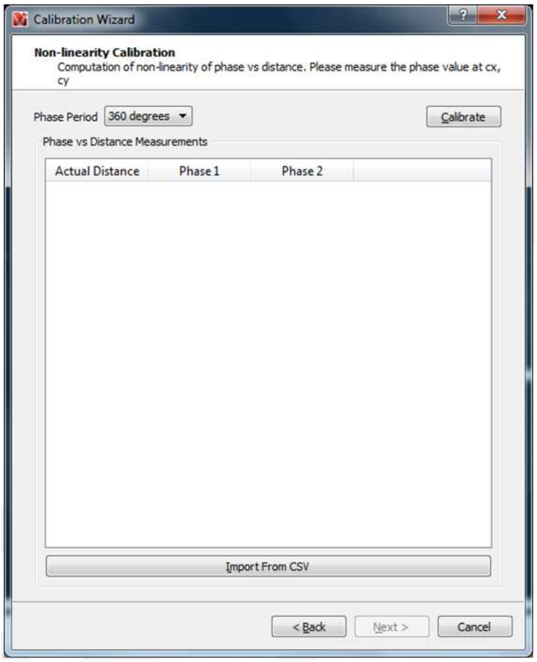
**Figure 29. Crosstalk Calibration Window**

### 4.4 Nonlinearity Calibration

For best accuracy, the relatively small nonlinear relationship that exists between the raw phase measurement and actual depth must be taken into consideration. Such relationship requires taking small steps between the starting position, and the end position, such that the travel spans the full 2π phase interval. Voxel Viewer can take in calibration data in *.csv format collected at calibrated optical center ($c_x$, $c_y$), in the following form:

<Actual Distance>, <Phase1>, <Phase2>

<Actual Distance> is the actual distance, <Phase1> is the phase measured at the first modulation frequency, and <Phase2> is phase measured at the second modulation frequency. These two modulation frequencies work together in the de-aliasing operation.



Calibration Wizard

**Non-linearity Calibration**

Computation of non-linearity of phase vs distance. Please measure the phase value at cx, cy

Phase Period: 360 degrees

Phase vs Distance Measurements

| Actual Distance | Phase 1 | Phase 2 |
|---|---|---|

**Figure 30. Frequency Calibration Window**

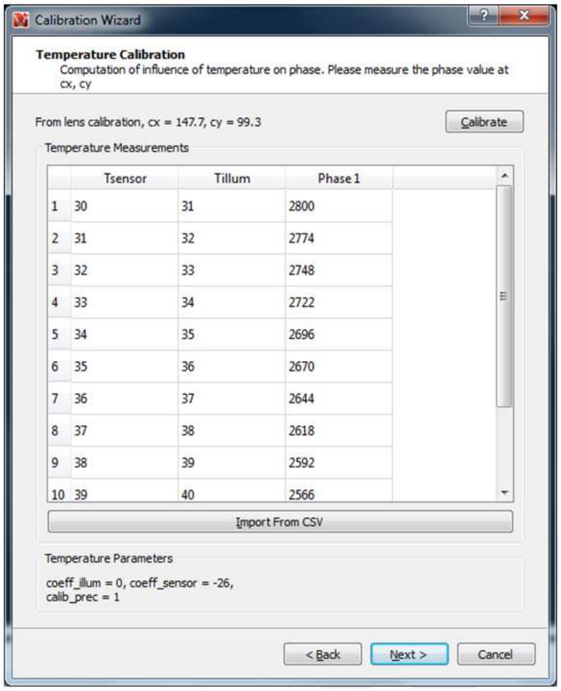
### 4.5 Temperature Calibration

The goal of temperature calibration is to derive a set of curve-fitting coefficients used in compensating phase drift due to temperature. Normally temperature calibration only needs to be done once per design. A thermal chamber is required to collect temperature-vs-phase data into a *.csv file format. Then the *.csv file is imported into Voxel Viewer in this step to compute the compensation coefficients. Each row of the *.csv file must contain three numbers separated by comma, with the following meaning:

<Tsensor>, <Tillum>, <Measured Phase>

<Tsensor> is the imaging sensor's temperature; <Tillum> is the temperature of the illumination driver, both in °C. Typically the data is collected in about 1°C intervals.

Once the *.csv file is imported, click **Calibrate**, and the computed temperature coefficients will be displayed at the bottom of the window, as is shown in Figure 31.



**Temperature Calibration**
Computation of influence of temperature on phase. Please measure the phase value at cx, cy

From lens calibration, cx = 147.7, cy = 99.3

Temperature Measurements

| | Tsensor | Tillum | Phase 1 |
|---|---|---|---|
| 1 | 30 | 31 | 2800 |
| 2 | 31 | 32 | 2774 |
| 3 | 32 | 33 | 2748 |
| 4 | 33 | 34 | 2722 |
| 5 | 34 | 35 | 2696 |
| 6 | 35 | 36 | 2670 |
| 7 | 36 | 37 | 2644 |
| 8 | 37 | 38 | 2618 |
| 9 | 38 | 39 | 2592 |
| 10 | 39 | 40 | 2566 |

Temperature Parameters

coeff_illum = 0, coeff_sensor = -26,
calib_prec = 1

**Figure 31. Temperature Calibration Window**

## 4.6 Common Phase Calibration

Common phase calibration assumes the phase-to-depth relationship is linearized by other calibration by the time it runs. The conversion of phase to depth can then be summarized by the linear equation:

$$d_i = \frac{c}{2f\eta}(\varphi_i - \varphi_{0i}) + d_{0i} \tag{2}$$

The common phase calibration correlates measured phase with actual depth at a single point ($\varphi_{0i}$, $d_{0i}$). This is done for all pixels indexed by *i*. Only the center pixel is correlated, and the average of 200 samples (default) is used.

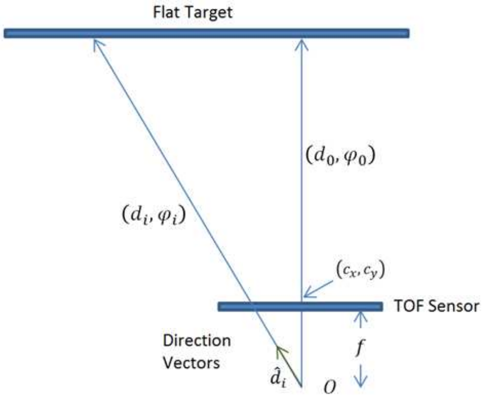
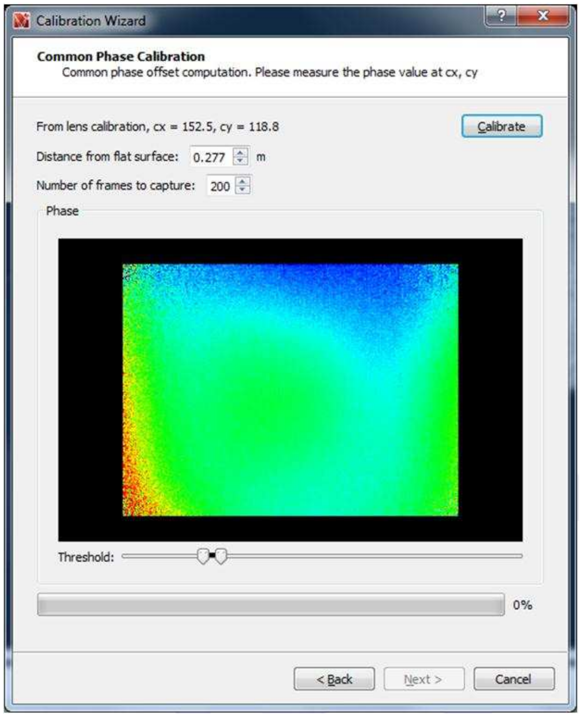


**Figure 32. Common Phase Calibration Setup**



Common Phase Calibration

Common phase offset computation. Please measure the phase value at cx, cy

From lens calibration, cx = 152.5, cy = 118.8

Distance from flat surface: 0.277 m

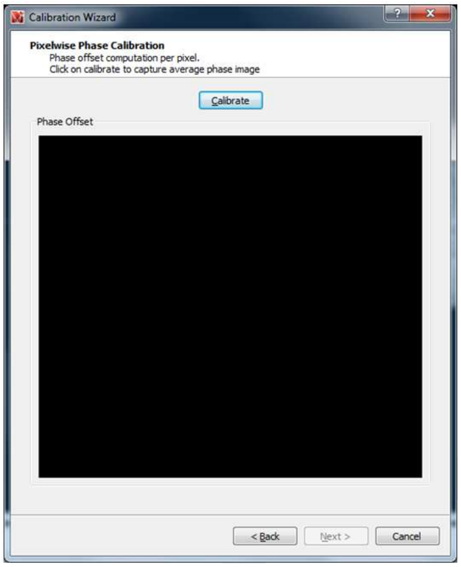
Number of frames to capture: 200

**Figure 33. Frame Capture for Common Phase Calibration**

## *4.7 Pixel-Wise Calibration*

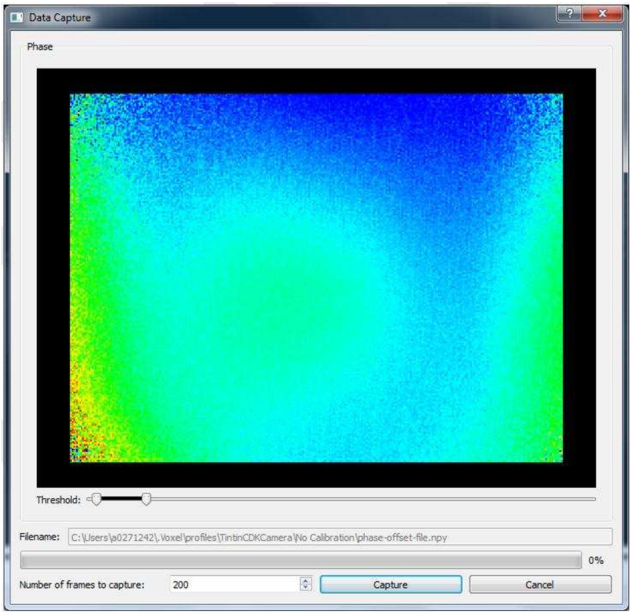
For high-precision applications, pixel-to-pixel calibration is recommended. This calibration step attempts to overcome the finer pixel-to-pixel variation. Since the variation is compensated for each pixel, this calibration requires a larger storage space on the EEPROM; and only one copy of the pixel variation map can be stored. In Section 4.8, the entire calibration sequence is summarized, and one will see that only one profile can have pixel-wise calibration stored to EEPROM.

To calibrate, point the camera at a flat wall where the wall covers the entire field-of-view. Figure 34 shows the pixel-wise calibration window. Clicking the Calibrate button in the calibration window will bring up the frame capture window, which will sample and average 200 frames. It is critical to keep the camera still during this time.



**Pixelwise Phase Calibration**
Phase offset computation per pixel.
Click on calibrate to capture average phase image

**Figure 34. Pixel-Wise Calibration Window**



Filename: C:\Users\a0271242\.Voxel\profiles\TintinCDKCamera\No Calibration\phase-offset-file.npy

0%

Number of frames to capture: 200

**Figure 35. Frame Capture Window**

## 4.8 Profiles and Calibration

Voxel Viewer comes with 5 built-in operating profiles:

- Lens Only
- Short Range
- Long Range
- High Ambient
- No Calibration

Typical calibration procedure involves creating several hardware profiles that are stored on the camera. These typically consist of:

**LensOnlyHW** – starts with **Lens Only** profile; inherits from **Lens Only**, calibrates for lens distortion.

**LongRangeHW** – starts with **Long Range** profile, inherits from **LensOnlyHW**; calibrates for frequency, temperature, common phase, and pixel-wise variations.

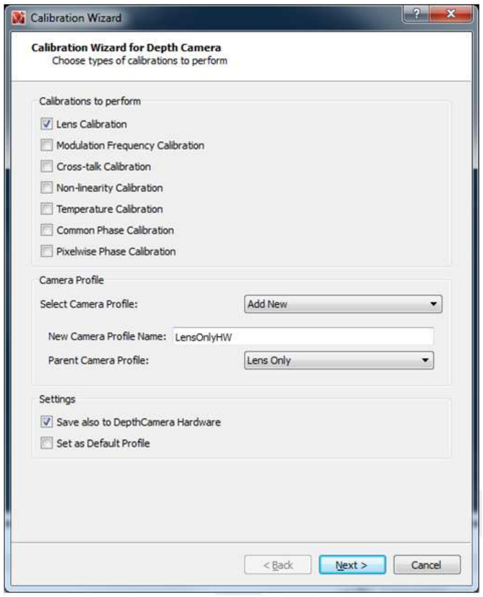
**ShortRangeHW** – starts with **Short Range** profile; inherits from **LongRangeHW**; calibrates for common phase only.

**HiAmbientHW** – starts with **High Ambient** profile; inherits from **LongRangeHW**; calibrates from common phase only.

Note that a profile can inherit the calibration results from another profile. This turned out to be an efficient mechanism of saving EEPROM space. The calibration detail is described in the following four steps:

**Step 1: Create LensOnlyHW Profile**

With the current operating profile set at **Lens Only** (**Settings** → **Camera Profiles** → **Lens Only**), bring up the Calibration Wizard (**Settings** → **Edit Profiles**), check the Lens Calibration checkbox, uncheck all the other boxes. Enter a new profile name, “LensOnlyHW” with **Parent Camera Profile** set to **Lens Only**. Check **Save also to DepthCamera Hardware** if you want to save the calibration parameters to the EEPROM. Click **Next** and follow the instruction listed in Section 4.1. The calibration distance should be far enough to see the entire checkerboard pattern (A2 format), yet close enough to fill the image as much as possible. When this step is done, two new profiles will be added: **LensOnlyHW** is the local cached copy of the lens calibration, and **LensOnlyHW (HW)** is the one stored onboard the camera.



Calibration Wizard

Calibration Wizard for Depth Camera
Choose types of calibrations to perform

Calibrations to perform
☑ Lens Calibration
□ Modulation Frequency Calibration
□ Cross-talk Calibration
□ Non-linearity Calibration
□ Temperature Calibration
□ Common Phase Calibration
□ Pixelwise Phase Calibration

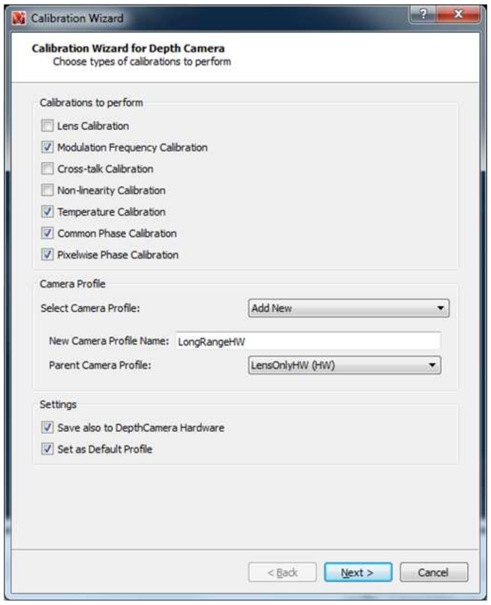
Camera Profile
Select Camera Profile: Add New
New Camera Profile Name: LensOnlyHW
Parent Camera Profile: Lens Only

Settings
☑ Save also to DepthCamera Hardware
□ Set as Default Profile

**Figure 36. LensOnlyHW Profile Settings**

### Step 2: Create LongRangeHW Profile

With the current operating profile set at **Long Range (Settings → Camera Profiles → Long Range)**, bring up the Calibration Wizard (**Settings → Edit Profiles**), configure the checkboxes as in Figure 37. The profile to save to is “LongRangeHW”. This profile inherits the lens calibration from the **LensOnlyHW (HW)** profile, and has several other calibration types included.



Calibration Wizard

**Calibration Wizard for Depth Camera**
Choose types of calibrations to perform

Calibrations to perform

- □ Lens Calibration
- ☑ Modulation Frequency Calibration
- □ Cross-talk Calibration
- □ Non-linearity Calibration
- ☑ Temperature Calibration
- ☑ Common Phase Calibration
- ☑ Pixelwise Phase Calibration

Camera Profile

Select Camera Profile: Add New

New Camera Profile Name: LongRangeHW

Parent Camera Profile: LensOnlyHW (HW)

Settings

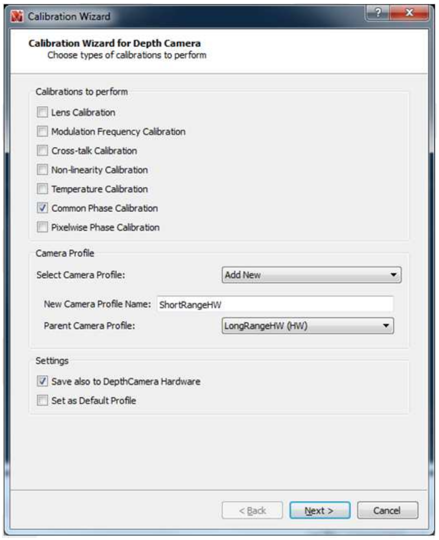
- ☑ Save also to DepthCamera Hardware
- ☑ Set as Default Profile

**Figure 37. LongRangeHW Profile Settings**

Click Next, and each of the checked calibration steps will be executed in sequence. The detail of each calibration step can be found in Section 4.2, Section 4.5, Section 4.6, and Section 4.7. After the entire sequence is completed, two more profiles will be created: **LongRangeHW**, which is the local cache, and **LongRangeHW (HW)**, which is stored on the camera. As one will see later, the **LongRangeHW (HW)** profile will be the parent profile for several other profiles. The **LongRangeHW** is also set as the default profile.

### Step 3: Create ShortRangeHW Profile

With the current operating profile set at **Short Range (Settings → Camera Profiles → Short Range)**, bring up the Calibration Wizard (**Settings->Edit Profiles**), configure the checkboxes as in Figure 38. The profile to save to is “ShortRangeHW”. This profile inherits from the **LongRangeHW (HW)** profile, and has only the **Common Phase Calibration** included, because **ShortRangeHW** and **ShortRangeHW (HW)** will inherit the results of other calibration types from the **LongRangeHW (HW)** profile. Click **Next** and proceed with the Common Phase Calibration described in Section 4.6. When this step is completed, two more profiles, **ShortRangeHW** and **ShortRangeHW (HW)** will be created.



Calibration Wizard

Calibration Wizard for Depth Camera
Choose types of calibrations to perform

Calibrations to perform
- □ Lens Calibration
- □ Modulation Frequency Calibration
- □ Cross-talk Calibration
- □ Non-linearity Calibration
- □ Temperature Calibration
- ☑ Common Phase Calibration
- □ Pixelwise Phase Calibration

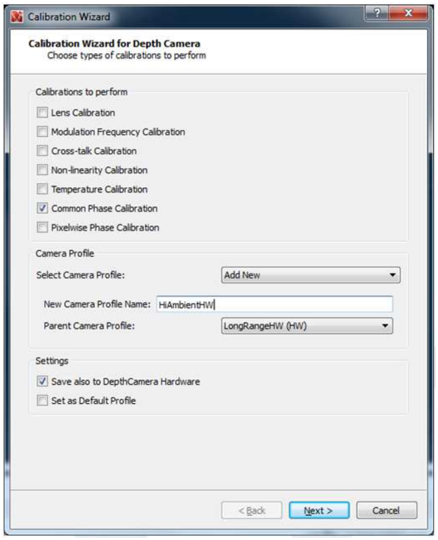
Camera Profile
Select Camera Profile: Add New
New Camera Profile Name: ShortRangeHW
Parent Camera Profile: LongRangeHW (HW)

Settings
- ☑ Save also to DepthCamera Hardware
- □ Set as Default Profile

**Figure 38. ShortRangeHW Profile Settings**

### Step 4: Create HiAmbientHW Profile

With the current operating profile set at **High Ambient (Settings → Camera Profiles → High Ambient)**, bring up the Calibration Wizard (**Settings → Edit Profiles**), configure the checkboxes as in Figure 39. The profile to save to is “HiAmbientHW”. This profile also inherits from the **LongRangeHW (HW)**. Click **Next** and proceed with Common Phase Calibration as before. When this step is completed, **HiAmbientHW** and **HiAmbientHW (HW)** profiles are created.



Calibration Wizard

Calibration Wizard for Depth Camera
Choose types of calibrations to perform

Calibrations to perform
- □ Lens Calibration
- □ Modulation Frequency Calibration
- □ Cross-talk Calibration
- □ Non-linearity Calibration
- □ Temperature Calibration
- ☑ Common Phase Calibration
- □ Pixelwise Phase Calibration

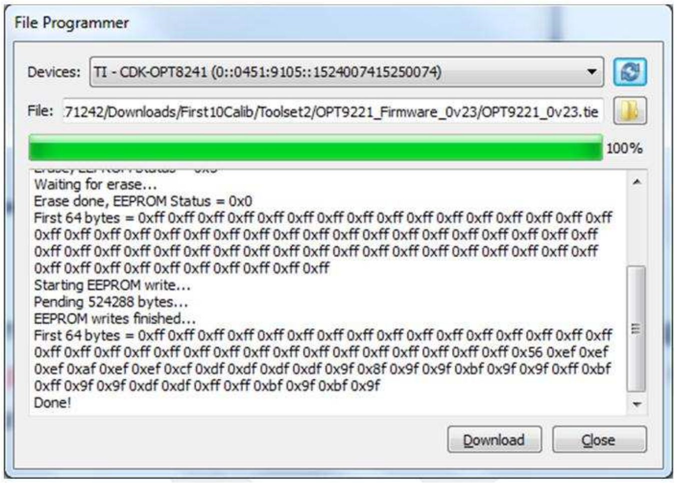
Camera Profile
Select Camera Profile: Add New
New Camera Profile Name: HiAmbientHW
Parent Camera Profile: LongRangeHW (HW)

Settings
- ☑ Save also to DepthCamera Hardware
- □ Set as Default Profile

**Figure 39. High Ambient Profile Settings**

## 5 TFC Programming

Voxel Viewer is also used to update the TFC program by selecting **Files** → **Open Programmer**. The top line displays and selects the camera being programmed. The second line displays and selects the *.tie or *.tip files to be programmed. Click on the folder icon to select the *.tie or *.tip file and click **Download**.


File Programmer
Devices: TI - CDK-OPT8241 (0::0451:9105::1524007415250074)
File: 71242/Downloads/First10Calib/Toolset2/OPT9221_Firmware_0v23/OPT9221_0v23.tie
100%
Download
Close

```
Waiting for erase...
Erase done, EEPROM Status = 0x0
First 64 bytes = 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
Starting EEPROM write...
Pending 524288 bytes...
EEPROM writes finished...
First 64 bytes = 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0x56 0xef 0xef
0xef 0xaf 0xef 0xef 0xcf 0xdf 0xdf 0xdf 0xdf 0x9f 0x8f 0x9f 0x9f 0xbf 0x9f 0x9f 0xff 0xbf
0xff 0x9f 0x9f 0xdf 0xdf 0xff 0xff 0xbf 0x9f 0xbf 0x9f
Done!
```

**Figure 40. TFC Programmer**

During programming, a series of messages will appear indicating that programming is in progress. Do not turn off the power at this point! When the programming is complete, a message "Done!" will appear at the end. Now disconnect the camera by **Files** → **Disconnect Depth Camera**. Cycle the camera power and reconnect using **Files** → **Connect Depth Camera**.

## 6 Summary

This document provided a brief introduction to the TI 3D Time-of-Flight technologies and the available camera development kits; next it described the Voxel Viewer, the visualization tool for these camera development kits. First the Voxel Viewer user interface was described, which included several view ports showing various 3D data types, and many windows displaying important data and parameters, giving users tremendous flexibility. Then the basic operation of the Voxel Viewer was explained, which included selection of different data to visualize, setting and saving of parameters, setting of watch points and debugging with logs. Finally, various types of calibration were discussed, the calibration procedure described, and TFC programming explained.

## 7 References

- *Time of Flight Camera: An Introduction* (SLOA190)
- *Introduction to the Time-of-Flight (ToF) System Design* (SBAU219)
- *Filtering for 3-D Time-of-Flight Sensors*
- TI 3-D Time-of-Flight Product Page
- Voxel SDK download
- OpenCV camera calibration tutorial

# IMPORTANT NOTICE

| **Products** | | **Applications** | |
|---|---|---|---|
| Audio | www.ti.com/audio | Automotive and Transportation | www.ti.com/automotive |
| Amplifiers | amplifier.ti.com | Communications and Telecom | www.ti.com/communications |
| Data Converters | dataconverter.ti.com | Computers and Peripherals | www.ti.com/computers |
| DLP® Products | www.dlp.com | Consumer Electronics | www.ti.com/consumer-apps |
| DSP | dsp.ti.com | Energy and Lighting | www.ti.com/energy |
| Clocks and Timers | www.ti.com/clocks | Industrial | www.ti.com/industrial |
| Interface | interface.ti.com | Medical | www.ti.com/medical |
| Logic | logic.ti.com | Security | www.ti.com/security |
| Power Mgmt | power.ti.com | Space, Avionics and Defense | www.ti.com/space-avionics-defense |
| Microcontrollers | microcontroller.ti.com | Video and Imaging | www.ti.com/video |
| RFID | www.ti-rfid.com | | |
| OMAP Applications Processors | www.ti.com/omap | **TI E2E Community** | e2e.ti.com |
| Wireless Connectivity | www.ti.com/wirelessconnectivity | | |

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265