SoftKinetic™

The Interface is You

DepthSense Viewer User Guide

SoftKinetic

DepthSense Viewer User Guide

SoftKinetic

Publication date May 23, 2013

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Welcome to the DepthSense Viewer v1.0 User Guide. This user guide provides detailed explanations on the DepthSense Viewer, how to use it and the philosophy behind it.

For more informations on DepthSense SDK please refer to the DepthSense SDK Developer Guide.

Your comments on DepthSense Viewer and this documentation are welcome and can be sent to `<support@softkinetic.com>`.

About DepthSense Viewer

The DepthSense Viewer is a demonstration tool to show DepthSense SDK capabilities as well as its mechanisms.

Showing DepthSense SDK mechanisms here means showing how it has to be used. Whereas some viewers would provide the user with an interface doing everything automatically for him, DepthSense Viewer requires the user to perform the steps he would have to do when using the DepthSense SDK.

This can at first look a bit forbidding, but the aim of the DepthSense Viewer is to help the user to get familiarized with the DepthSense SDK: doing something wrong programmatically can result in hours of debugging, but some typical errors can be learned using DepthSense Viewer, thus saving time afterward.

Frequently used terms

Device	a full DepthSense device, including depth camera, color camera, microphone, etc.
Node	a sub-device of a DepthSense device, for example a depth camera, color camera or microphone.
Buffer	a representation of a node's data. For example to view a 2D representation of the depth produced by the depth node, the depth buffer of that particular node must be displayed.
Context	Everthing in DepthSense SDK is accessed through a context. To stream a node, that node must be registered to a context, then the streaming of all registered nodes will be started using the context. To change the settings of a node, the context must take control of the node, then changes can be applied. The DepthSense Viewer can in fact be seen as a single context.

Chapter 2. Getting started

Starting DepthSense Viewer

The DepthSense Viewer can be launched like any application and accepts some command line options. To see the full list of options please refer to Appendix A, *Command line options*.

Multiple instances of the DepthSense Viewer can be launched at the same time, although only one instance can take full control of a given node.

DepthSense Viewer main screen

Once DepthSense Viewer is launched, the main screen is displayed. The main screen can be divided in four parts:

- the device list panel on the left;

- the information/configuration panel on the right;

- the central area being there to display buffers;

- and the context control tool bar on top.
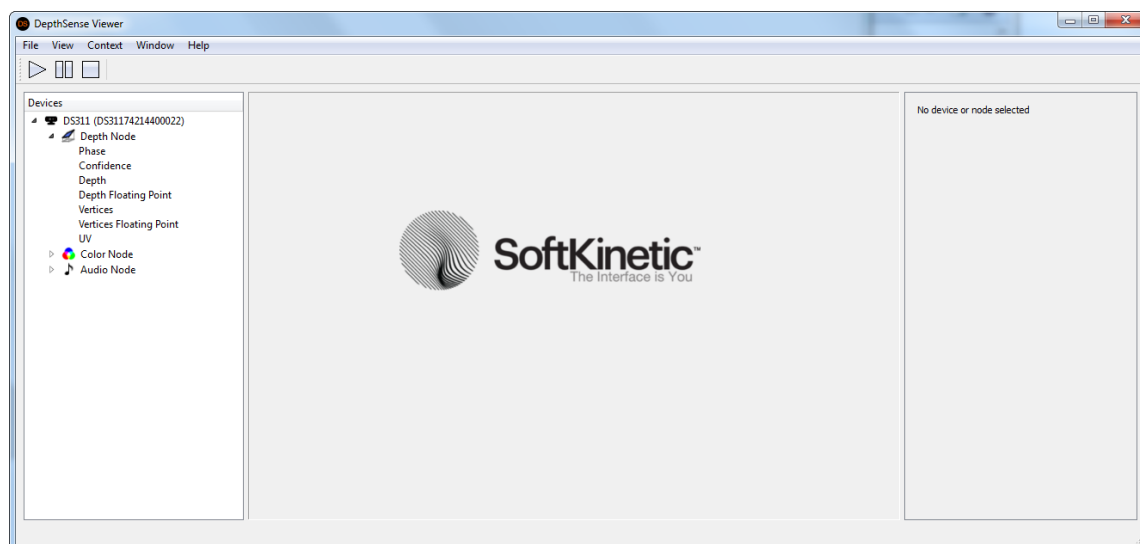
Devices list

The left panel displays the list of connected devices in a form of a tree view. Each device is represented by a camera icon, the model of the camera and the serial number (in parenthesis). Selecting a device will update the right-hand side panel with informations on that device.

Each device entry can be expanded, showing the different nodes of that device. DepthSense cameras usually contain 3 different nodes : a depth node, a color node and an audio node. Selecting a node will also update the panel on the right with informations, as well as with some configuration controls depending on the type of node.

Each node can also be expanded, showing the different exposed buffers of this particular node. Selecting a buffer will display the same things as if its parent node was selected. To display the content of a buffer, it must be double clicked or drag-and-dropped in the central display area.

Display area

The central area is there to display different buffers. Just after DepthSense Viewer is started, this area is empty. If one wants to display the content of a buffer, that buffer must be added using the left panel.

Each time a buffer is added to the central area, a new sub-window is opened. Each buffer can only be added once (if a buffer is added twice, nothing will happen) and can be removed by closing the sub-window. Apart from the buffer content, this sub-window also contains different informations :

- The *frame rate* is the number of distinct frames displayed per second.

- The *event rate* is the number of new frame event received per second from the DepthSense SDK. The event rate and the frame rate are the same most of the time, but for example, if the context is paused, no new frame will be displayed, but the events will still be received.

- The *paint rate* is the number of times the display is refreshed. If for example the mouse goes over the viewport, this paint rate will increase above the frame rate.

- The *time of capture* is the time at witch the device produced the data.

- The *time of arrival* is the time at witch DepthSense SDK receives the data from the USB port.

- The *time of event* is the time at witch DepthSense Viewer receives a new frame event from the DepthSense SDK.

- The *time of display* is the time at witch the buffer is rendered with OpenGL in the DepthSense Viewer.

- The *dropped frames* field gives the number of frames that were dropped between this frame and the previously received frame.

- The *total dropped frames* field gives the number of frames that were dropped since DepthSense Viewer is running.

- The *acceleration* field is displayed for the depth node's buffers and gives the acceleration vector of the camera.

The different sub-windows can be organized as one wishes inside the display area and sub-windows can be cascaded or tiled using the corresponding entries in the *Windows* menu.

Context control

The notion of context is central in the DepthSense SDK. When using DepthSense SDK, the different nodes control is done through a context. To stream a node, that node must be registered to a context and the `Context` class member `startNodes()` must be called;

As said previously, the DepthSense Viewer was designed to forces the user to have the way DepthSense SDK is working in mind: the viewer itself can be seen as a context. The tool bar in the upper part of the viewer gives access to this single context actions. From left to right :

Table 2.1. Toolbar buttons

Button	DepthSense SDK `Context` method	Description
Play	`startNodes()`	Starts the streaming of all nodes registered to the context.
Pause	/	Doesn't perform any action by itself, but once it is pressed, the DepthSense Viewer will ignore all `onNewFrame` events, thus freezing the display.
Stop	`stopNodes()`	stops the streaming of all nodes registered to the context.
Register node	`registerNode()` or `unregisterNode()`	Registers or unregisters the selected node to the context depending on the state of the toggle button.
Control node	`requestControl()` or `releaseControl()`	Try to give control over the node to the context or to release control depending on the state of the toggle button. It will fail if another context (another program) has already the control.

The equivalent actions can be found in the *Context* menu.

Information and configuration panel

The panel on the right of the display area is there to display informations and configuration controls related to the selected device or node (or the parent node of the selected buffer). Devices cannot be configured directly so only their informations will be displayed.

On the other hand, selecting a node will provide the user with all the configuration control available for that node.

4

Figure 2.1. Configuration panel examples

The configuration panel is divided vertically in three parts. The top part contains information that cannot be modified directly by the user. It can however be modified indirectly by modifying a property that has an impact on that information. For example, changing from close mode to long range mode changes the value of the range information.

The second part contains all the properties of the selected node. Those grayed out are read only, generally meaning that the context did not yet take the control over that node. Control can be taken by pressing the *Control Node* button in the tool bar. Modifying any of the check boxes properties will send the modification directly to the node. For other fields, enter must be pressed or the focus must be moved to another property.

The third part contains the available configurations of the selected node. Here again if the context does not have the control over the node, the widgets will all be disabled and control must be taken over the node to change them. Once the desired fields of the configuration have been selected, the *Set Configuration* button must be pressed to send the changes to the selected node.

Chapter 3. Streaming nodes

Basically, streaming the node is pretty straightforward, one just needs to register the node, drag and drop the buffers to visualize and click *Start Nodes*. Of course for configuration changes, control over the node must be taken.

As said before, DepthSense Viewer forces the user to perform all the steps he would have to perform with the DepthSense SDK. However there is an exception : when using the DepthSense SDK, to stream a particular buffer, one needs to enable the corresponding property. For example to stream the Depth buffer, the enableDepthMap property must be enabled. To make things a little bit more comfortable, when one opens a buffer window, DepthSense Viewer will automatically enable the corresponding property (except when multiple buffers are needed, like the UV buffer for example, in this case only the main property is enabled).

Once the buffer is open, unchecking the property will of course disable the streaming of the corresponding buffer, even if the window is opened.

Another point worth mentioning is that once the context is streaming, new nodes can be registered, new buffers added and they will be streaming directly.

Pinpoint tool

A pinpoint tool is available for all 2D buffers (all except Vertices and UV which are 3D buffers). When moving the mouse over the display area, a cross is superimposed over the buffer. Some informations concerning the pointed pixel is then displayed underneath.

The information displayed depends on the type of buffer : the color buffer will display RGB information, depth buffer will display the depth in mm, etc.

Region Of Interrest (ROI)

Some heavy computations can be done on a part of the frame, the region of interrest. To select the ROI, one only needs to press the mouse button on a 2D buffer, move it to define the ROI and release the button when done. Double-clicking removes the ROI for a given node.

A ROI can be defined for each node, and will be displayed on all 2D buffers of the corresponding node.

Depth node

Depth node properties

First of all, the depth node has nine properties controlling the different buffers streaming : *enablePhaseMap*, *enableConfidenceMap*, *enableDepthMap*, *enableDepthMapFloatingPoint*, *enableVer-*

tices, *enableVerticesFloatingPoint*, *enableVerticesPlanar*, *enableVerticesPlanarFloatingPoint* and *enableUvMap*. These properties are checkboxes that enable streaming of the corresponding buffer when checked.

enableAccelerometer enables the accelerometer sampling. If checked, the acceleration vector will be displayed in all the depth node buffer windows. Otherwise, a zero vector is displayed.

enableCloseMode controls the range of the camera. If checked, the camera will work in close range mode, otherwise in long range mode.

enableSaturation controls whether the saturation detection is used. Saturated pixels will in that case have a different color than low confidence pixels. Otherwise saturated and low confidence pixels cannot be differentiated.

_noFan is a property that should be used with extreme caution, if at all. If turned on, the effect is to turn off the fan while the temperature is low enough. The problem is that the temperature is monitored in software, and if for any reason the DepthSense SDK terminates unexpectedly, the streaming will continue without fan and without temperature monitoring, at best damaging the camera (you can use your imagination for the worst case).

Depth node configuration

The *format* configuration field defines the capture resolution.

The *frame rate* configuration field defines the rate at which the camera will capture new frames. For the moment only 25fps and 30fps are available.

Depth node buffers

Buffer	Related property	Description
Phase	*enablePhaseMap*	Displays the radial distance to the camera. Units are such as 0 represents the camera and 32767 the range of the camera. Units are thus different in close or long range mode.
Confidence	*enableConfidenceMap*	Displays the confidence of each pixel, 0 (black) being no confidence at all.
Depth	*enableDepthMap*	Displays the Cartesian Z coordinate. Units are in mm.
Depth Floating Point	*enableDepthMapFloatingPoint*	Displays the Cartesian Z coordinate. Units are in m.
Vertices	*enableVertices*	Displays a 3D view of all pixels. Units are in mm.
Vertices Floating Point	*enableVerticesFloatingPoint*	Displays a 3D view of all pixels. Units are in m.
Vertices Planar	*enableVerticesPlanar*	Displays a 3D view of all pixels. Units are in mm.

Buffer	Related property	Description
Vertices Planar Floating Point	*enableVerticesPlanarFloatingPoint*	Displays a 3D view of all pixels. Units are in m.
UV	*enableUvMap (enableVerticesFloatingPoint, enableColorMap)*	Displays a 3D view of all pixels colored with the color map. To be able to see this one must be sure that two additional properties are enabled : enableVerticesFloatingPoint and enableColorMap of the camera's color node. The color node must be streaming as well.
Standard Deviation	*(enableDepthMapFloatingPoint)*	Displays a 2D view of all the standard variations contained in the slected ROI for this node.

Controlling the camera in 3D views

Initially, the camera is located at the same place as the physical camera and is looking towards the scene. Using the mouse, one can move the camera around the scene freely.

At the center of the scene, there is a red dot. This dot represents what the camera is looking at and from here on will be called taget.

Left-clicking on the window and moving the mouse up and down will move the camera latitudinally around the taget, whereas moving the camera left and right will move it longitudinally around that point.

Right-clicking the window while moving the mouse up and down will move the camera closer or away of the taget.

Left-clicking on the window while holding the CTRL key pressed and moving the mouse up, down, left or right will move the camera as well as the taget in the corresponding directions.

Right-clicking the window while holding the CTRL key pressed and moving the mouse up and down will move the camera as well as the taget forward and backward.

Left-clicking on the window while holding the ALT key pressed and moving the mouse up and down will tilt the camera (up/down rotation), whereas moving the camera left and right will pan the camera (left/right rotation). The taget will be moved so that it remains the center of focus of the camera and stays at the same distance.

Left-clicking on the window while holding the SHIFT key pressed will move the taget on the closest point of the scene clicked by the mouse. The camera will keep it's position but will rotate to look in the direction of the new taget.

Double-clicking on the display frame will move back the camera at its default location.

Predefined views

Five predefined views are ready to be used to move the camera at the desired position. The camera can be moved to these five positions using the corresponding toolbar buttons. The buttons show

the way the coordinates axes will look after the camera has been moved (according to the default values).

When starting the viewer for the first time these five positions are as follow:

- The camera is located at the origin of the Cartesian coordinates and is looking towards positive Z, in other words, at the same place as the physical camera;

- The camera is located on the left side of the scene (-1, 0, 0.5) and is looking to the right (positive X direction);

- The camera is located on the right side of the scene (1, 0, 0.5) and is looking to the left (negative X direction);

- The camera is located above the scene (0, 1, 0.5) and is looking down (positive Y direction);

- The camera is located bellow the scene (0, -1, 0.5) and is looking up (negative Y direction).

By default, the taget is located at (0, 0, 0.5) in all five views.

The five predefined views can be customized in the preferences dialog.

Depth node snapshot

To export the depth node data in a .csv file, you can use the *File > Export > Current Frame...* menu or the corresponding button in the toolbar. This button is only enabled if the selected depth node has been streaming and is now paused and no node properties/configurations have been changed since the last streaming.

When this button is pressed a file selection dialog appears to select the file to be exported.

The saved .csv file will contain the timestamp and acceleration information as well as all the enabled buffers content (phase, confidence, depth, vertices, etc.).

Color node

Color node properties

The *enableColorMap* property enables streaming of the the color buffer when checked.

The *enableCompressedData* property enables streaming of the compressed color buffer if checked (and if MJPEG compression type has been configured).

The *brightness* property controls the color camera brightness and ranges from -10 to +10.

The *contrast* property controls the color camera contrast and ranges from 1 to 32.

The *saturation* property controls the color camera saturation and ranges from 0 to 20.

The *hue* property controls the color camera hue and ranges from -5 to +5.

The *gamma* property controls the color camera gamma and ranges from 100 to 200.

The *whiteBalance* property controls the color camera white balance and takes the values 2800, 4650 and 6500.

The *sharpness* property controls the color camera sharpness and ranges from 0 to 10.

The *exposure* property controls the color camera manual exposure and ranges from 156 to 5000.

The *exposureAuto* property defines which type of exposure system the color camera is using. It can either be Manual or AperturePriority.

The *exposureAutoPriority* property defines whether exposureAuto has priority over the guaranteed frame rate or not.

The *whiteBalanceAuto* property enables the auto white balance of the color camera if set.

Color node configuration

The *format* configuration field defines the capture resolution.

The *compression* configuration field defines without compression is used (M-JPEG) or not (YUY2).

The *frame rate* configuration field defines the rate at which the camera will capture new frames.

The *PL freq* configuration field tells the camera what is the power line frequency in use. In most cases, 50Hz should be used with a frame rate of 25fps and 60Hz with 30fps.

Color buffer

Buffer	Related property	Description
Color	*enableColorMap*	Contains the color image captured by the color node.

Audio node

Audio node properties

The *inputMixerLevel* property controls the input volume. The value can range from 0.0 to 1.0.

The *mute* property mute the camera microphones if checked.

Audio node configuration

The *sampling rate* configuration field defines how much samples are recorded per second.

The *bits per sample* configuration field defines the size of the data used for one sample.

The *channels* configuration field defines the number of channels to be recorded.

Audio buffer

Buffer	Related property	Description
Audio	*none*	Contains the audio samples for duration of one second. When one second is reached, the following samples are drawn starting back at the start of the buffer.

Chapter 4. Preferences

Preferences can be reached through the View menu.

General preferences

Automatically save workspace on exit: if checked, the connected cameras configuration, registration state, configuration state and opened buffers will be saved on exit. The main window size and position are saved as well. *Default is OFF*.

The next time the application is launched, all these will be restored if the camera is connected (buffers sub-windows position and size included). If the camera is not connected on the next launch, nothing will happen until the saved camera is connected. Once connected, the camera will be configured automatically as it was.

Automatically expand devices: if checked, all connected devices will start expanded in the tree view on the left, showing their nodes. *Default is ON*.

Automatically expand devices: if checked, all connected devices nodes will start expanded in the tree view on the left, showing their buffers. *Default is ON*.

2D Views

Standard deviation number of frames: defines the number of frames the standard deviation is computed on. *Default is 100*.

ROI color: Color of the ROI rectangle displayed on 2D views. *Default is #3399FF*.

3D Views

Dots size: defines the dots diameter in all the 3D views. *Default is 2*.

Display triangles instead of dots in UV view: If checked, the UV 3D view is rendered using textured triangles instead of colored points. *Default is OFF*.

Display camera axes: If checked, the axes of the Cartesian coordinates centered on the physical camera are displayed (Red for the X axis, green for Y and blue for Z). *Default is ON*.

Display axes labels: If checked, X, Y and Z labels are added to the axes. *Default is OFF*.

Non mappable UV pixel color: Color used to display pixels with UV coordinates outside the [0,1] interval. Setting this color to black make them hardly visible as the background color is also black. *Default is Black*.
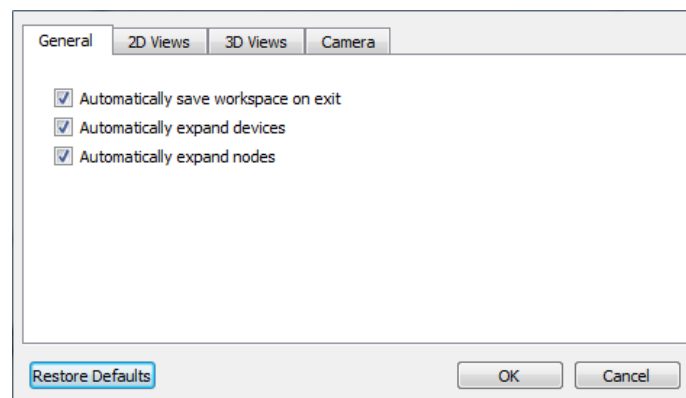
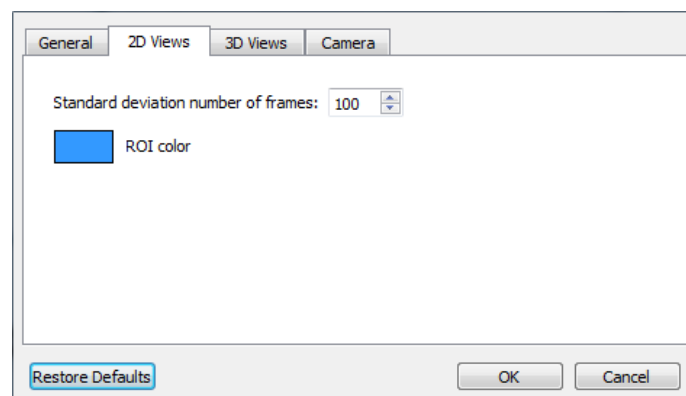*Background color*: 3D views background color. *Default is Black*.

Camera

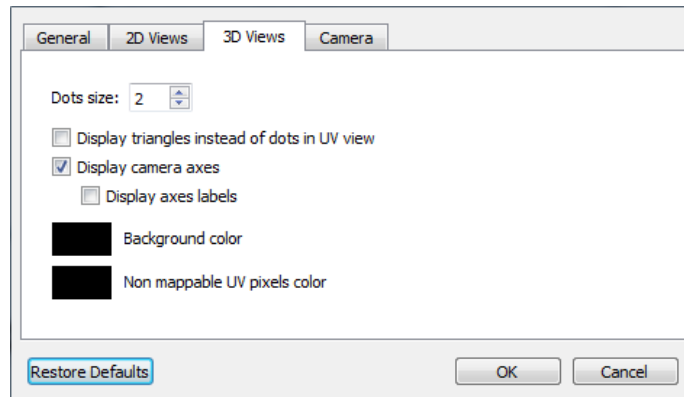This tab gives the possibility to change the preset camera views.

Default values:

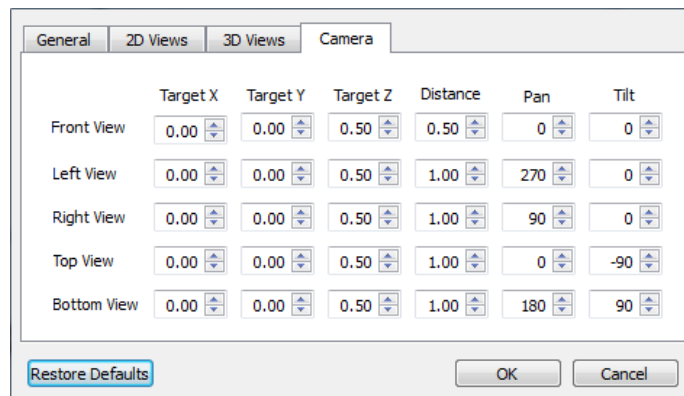|	Target X	Target Y	Target Z	Distance	Pan	Tilt
Front View	0	0	0	0.5	0	0
Left View	0	0	0.5	1	270	0
Right View	0	0	0.5	1	90	0
Top View	0	0	0.5	1	0	-90
Bottom View	0	0	0.5	1	180	90

Figure 4.1. General preferences

Figure 4.2. 3D views preferences

Figure 4.3. 3D views preferences

Figure 4.4. Camera preferences

Appendix A. Command line options

The command line options are shown in Table A.1, "Command line options".

Table A.1. Command line options

-h	--help	Print the command line help
-sa	--standalone	Use the depthsense driver in standalone mode
-dd	--disabledepth	Don't use depth nodes
-dc	--disablecolor	Don't use color nodes
-da	--disableaudio	Don't use audio nodes

Appendix B. DepthSense Viewer menu

File	Export > Current Frame...	Save the current depth node frames in a .csv file.
	Quit	Leaves the application.
View	Mirror view	If checked, all the buffers are rendered as mirror views, otherwise they are rendered as camera views.
	Preferences...	Open the preferences dialog.
Context	Start nodes	Starts the streaming of all nodes registered to the context.
	Pause nodes	Doesn't perform any action by itself, but once it is pressed, the DepthSense Viewer will ignore all onNewFrame events, thus freezing the display.
	Stop nodes	stops the streaming of all nodes registered to the context.
	Register node	Registers or unregisters the selected node to the context depending on the state of the toggle button.
	Control node	Try to give control over the node to the context or to release control depending on the state of the toggle button. It will fail if another context (another program) has already the control.
Windows	Cascade	Opened subwindows are cascaded.
	Tile	Opened subwindows are tiled.
Help	About...	About DepthSense Viewer