



# HTC Vive SRWorks SDK Guide

Release version: 0.8.0.2

©2017-2018 HTC Corporation. All Rights Reserved. HTC, the HTC logo, Vive, the Vive logo, and all other HTC product and services names are the trademarks or registered trademarks of HTC Corporation and its affiliates in the U.S. and other countries.

All other trademarks and service marks mentioned herein, including company names, product names, service names and logos, are the property of their respective owners and their use herein does not indicate an affiliation with, association with, or endorsement of or by HTC Corporation.

## Version notes

Release date	Version no.	Content
2018.11.08	0.8.0.2	Refer to the release notes for feature changes
2018.05.31	0.7.5.0	Refer to the release notes for feature changes
2018.03.02	0.7.0.0	Refer to the release notes for feature changes
2017.12.29	0.6.0.0	Refer to the release notes for feature changes
2017.11.30	0.5.0.0	Refer to the release notes for feature changes
2017.10.31	0.3.3.0	<p>Supported features:</p> <ul style="list-style-type: none"><li>• Stereo vision<ul style="list-style-type: none"><li>○ Real and virtual world alignment</li><li>○ Visual effects</li></ul></li><li>• Depth sensing</li><li>• Live 3D space reconstruction<ul style="list-style-type: none"><li>○ Mesh creation</li><li>○ Color texture</li><li>○ Physical collision</li><li>○ Plane detection</li><li>○ OBJ 3D model file saving</li></ul></li><li>• Demo/tutorial (with source code)</li></ul>

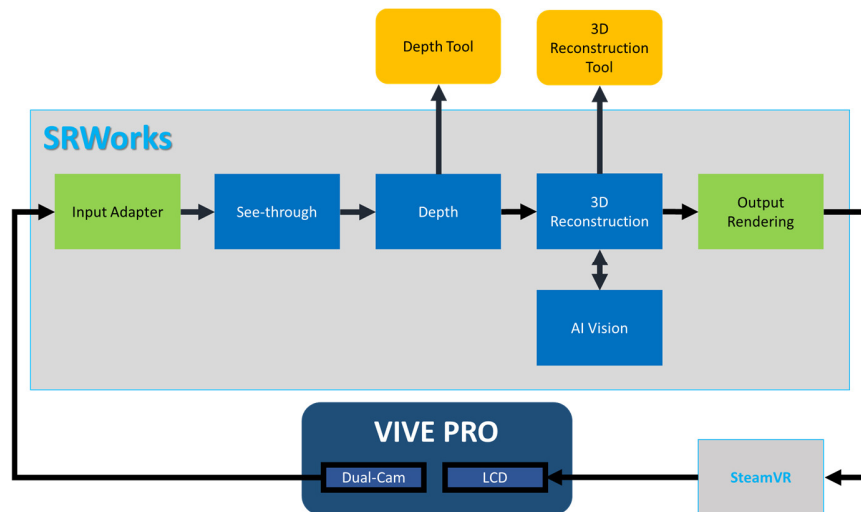
# Table of Contents

Version notes.....	3
About the Vive SRWorks SDK .....	5
Reality/virtuality interface through the dual camera .....	6
System requirements.....	7
Limitations.....	錯誤! 尚未定義書籤。
Basic preparations .....	7
SDK folder structure .....	8
Installing C sample code.....	9
Installing the Unity plugin.....	9
Opening a sample scene.....	錯誤! 尚未定義書籤。
Using SRWorks Unity API .....	9
Installing the Unreal Plugin .....	9
Known issues .....	9
Frequently asked questions.....	10

## About the Vive SRWorks SDK

The Vive SRWorks SDK expands the capabilities and potential uses of the see-through technology of the Vive VR system. With this, content developers can bring real world into VR.

In most cases, the view of the real world is blocked while wearing the headset. By using the dual camera of Vive Pro, the headset can receive input from the real world in stereo vision.



Images from the dual camera undergo processing through the following Vive SRWorks modules, after which the output can be rendered through Unity or Unreal and delivered to SteamVR:

- **See-through module.** This enables Vive Pro's dual camera's see-through function to view the actual environment outside VR. Furthermore, the module also provides material and texture effects.
- **Depth module.** This provides depth-sensing of objects in the real world. By using a third-party depth tool, developers can further add depth of field information, such as focal length and object distance, to their VR content.
- **AI Vision module.** This provides human or indoor objects (chair/table/floor/wall/ceiling/bed) segmentation information in the real world. This module can recognize 2D images and 3D scene for MR application.
- **3D reconstruction module.** This supports saving of 3D scenes as an .OBJ file. The module recreates scanned environments into 3D, including mesh and material assets, color textures, as well as detection of physical collisions and planes. If this module enables AI Vision relative function, it will export semantic objects (.obj) and extra-information (.xml) for plenty MR application. Furthermore, a third-party 3D reconstruction tool can be used to generate detailed 3D models from images captured by the dual camera or another photo equipment.

## Reality/virtuality interface through the dual camera

The Vive Pro headset's dual camera provides a stereo view of the real world, mimicking human vision via two lenses each with its own image sensor.

For developers planning to use see-through technology with their VR content, the dual camera offers two major benefits:

- **3D perception:** Because there are separate cameras used for the left and right eyes, 3D perception is possible when viewing the real world.
- **Depth sensing:** Depth information can be used for advanced developments, such as for body, hand, and face tracking. This needs further implementation and testing done by developers.

By using the SRWorks SDK, developers can create VR content that can simultaneously align the real world and VR. The see-through feature, supported by 3D reconstruction and depth sensing, offers untapped possibilities for developing next-generation VR content across genres – from games, entertainment, to education.



## System requirements

To develop applications in Unity using the Vive SRWorks SDK, the following minimum software and hardware requirements should be met:

Software requirements	<ul style="list-style-type: none"><li>• Windows 8.1 or later (64-bit)</li><li>• Unity or Unreal engine</li><li>• Steam and SteamVR (October 14 release or later)</li></ul>
Hardware requirements	<ul style="list-style-type: none"><li>• HTC Vive Pro VR system</li><li>• USB 3.0 port</li><li>• Nvidia GeForce GTX 970 graphics card</li></ul>

## Basic preparations

Before you develop applications with the SDK, you need to note the following considerations:

- The current depth sensing error rates are:
  - 1m: 3%, i.e.: +/- 3cm
  - 2m: 5%, i.e.: +/-10 cm
- The closer or farther the distance, the lower or higher the error rate. Based on the error rate, use linear interpolation/extrapolation to estimate your desired outcome.
- To prevent distortions under low light, use a plain color environment for the depth and 3D scan functions.

## SDK folder structure

Vive-SRWorks-*version*\

- **SRWorks Release Note- *version*.docx**
- **SRWorks SDK Guide- *version*.docx**
- C-Sample-*version*.zip\
  - **Getting Started with C API for SRWorks.docx**
  - **ViveSR.sln**: sample code solution file
  - model\
    - human\ : AI model for human
    - sceneSemantic\ : AI model for semantic scene
  - sample-vs2015\
    - **ViveSR\_VS2015\_SampleCode.vcxproj**: sample code project file
  - sample-ModuleAI\_Vision\_2D\
    - **sample-ModuleAI\_Vision\_2D.vcxproj**: sample code project file
  - sample-ModuleAI\_Vision\_3D\
    - **sample-ModuleAI\_Vision\_3D.vcxproj**: sample code project file
  - sample-vs2015\_CameraControl\
    - **CameraControl.vcxproj**: sample code project file
  - sample-vs2015\_ModuleDepthControl\
    - **sample-ModuleDepthControl.vcxproj**: sample code project file
  - x64\
    - Release\
      - **sample-vs2015.exe**: pre-built of C sample code
      - **sample-ModuleAI\_Vision\_2D.exe**: pre-built of C sample code
      - **sample-ModuleAI\_Vision\_3D.exe**: pre-built of C sample code
      - **sample-CameraControl.exe**: pre-built of C sample code
      - **sample-ModuleDepthControl.exe**: pre-built of C sample code
- Plugin-Unity-*version*.zip \
  - **Vive-SRWorks- *version* -Unity-Plugin.unzippackage**: SRWorks Unity plugin
  - **Getting Started with SRWorks in Unity.docx**
  - **SRWorks Unity API Reference.docx**
  - **SRWorks Unity Portal Guideline.docx**
- Plugin-Unreal-*version*.zip \
  - **Getting Started with SRWorks in Unreal.docx**
  - **Vive-SRWorks- *version* -Unreal-Plugin.zip**: SRWorks Unreal plugin
- Experience-Unity-*version*.zip \
  - **Vive-SRWorks-*version*-Unity-Experience.unzippackage**: Source code of SR Experience demo application including all features of the SDK.
  - **Getting Started with SRWorks Experience in Unity.docx**
- Experience-Unreal-*version*.zip \
  - **Getting Started with SRWorks Experience in Unreal.docx**
  - **Vive-SRWorks- *version* -Unreal-Experience.zip**: Sample code of SR Experience



- Build-Unity-version.zip \
  - Experience\
    - **Vive SRWorks Demo.exe**: pre-built of SR Experience
  - Plugin\
    - **Vive SRWorks.exe**: pre-built of plugin's sample scene

## Installing C sample code

Please refer to **Getting Started with C API for SRWorks.docx**.

## Installing the Unity plugin

Please refer to **Getting Started with SRWorks in Unity.docx** and **Getting Started with SRWorks Experience in Unity.docx**.

## Using SRWorks Unity API

After installing the SDK and trying the demo program, you can start developing your content. For details, refer to the **SRWorks Unity API Reference.docx**

## Installing the Unreal Plugin

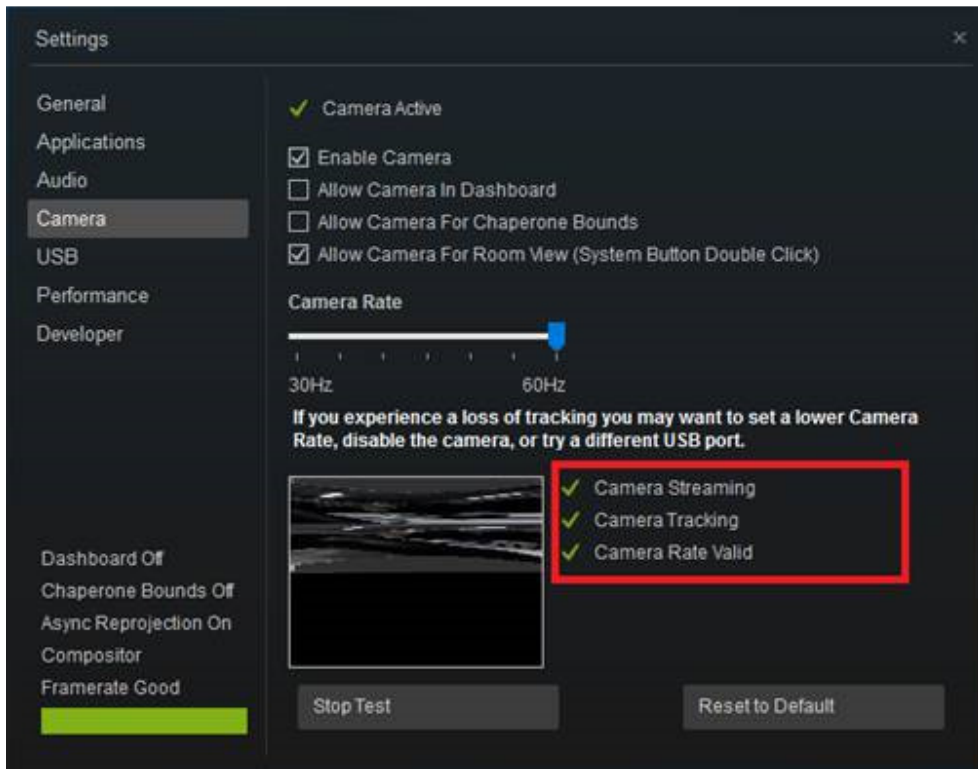
Please refer to **Getting Started with SRWorks in Unreal.docx** and **Getting Started with SRWorks Experience in Unreal.docx**.

## Known issues

## Frequently asked questions

*How can I check if my Vive Pro is compatible with the SRWorks SDK?*

Open SteamVR, and then click **Settings > Camera**. Select **Test Camera Rate**, and check if you see the following 3 green checkmarks (as illustrated below):



*Is there any sample code for developers to help them learn about the SDK?*

Yes, a Unity tutorial application along with its source code is provided to demonstrate the functionalities of the Vive SRWorks SDK. You can find this in: SRWorksSDK\Vive-SRWorks-version\Experience-Unity- version\Vive-SRWorks-version-Unity-Experience.unitypackage.

*What function in SRWorks currently uses the depth image?*

The 3D reconstruction module.

*Can controller functions in the real world align with VR?*

Yes. For example, when the Vive controller is being viewed with the see-through camera, the controller can still interact in VR mode as well.

*What could mitigate the concerns about the camera's VGA resolution?*

Filters can be applied to the video generated for more visually pleasing content.

*What is the latency rate from the camera to the headset display?*

Typical latency is 0.1 seconds, and may vary depending on the computer hardware and loading speed. Latency is currently being fine-tuned.