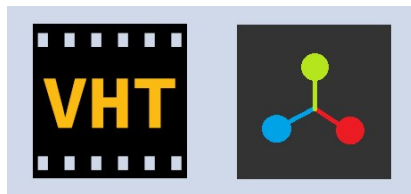


VHT tools for Blender

User Guide

v1.0



virtualhometheater.com

Table of Contents

Introduction.....	2
Concept Preview.....	3
Use Case Examples.....	4
Software Requirements.....	4
Hardware Requirements.....	5
Highlighted Features.....	5
Installation.....	5
Prerequisite Settings.....	6
Blender Preparation.....	7
Quick First Use.....	8
VHT Tools Interface Description.....	9
Cameras.....	9
Add VR stereoscopic camera:.....	10
EQD180.....	10
REL9mm.....	10
VR180.....	10
VR360.....	10
Create VR stereoscopic window:.....	10
EQD180 SbS crosseye.....	10
VR180 SbS crosseye.....	10
EQD360 SbS crosseye.....	10
Blender to VHT messaging bridge:.....	11
Launch VR session.....	11
<i>End VR session</i>	11
Rendering.....	12
Start to render IMAGE:.....	12
Quick at 16 samples + den.....	12
at 1024 samples + den.....	12
as SET.....	12
Start to render VIDEO:.....	13
Quick at 25% + 16 sam + den.....	13
at 1024 samples + den.....	13
as SET.....	13
VHT Control:.....	14
GUI_repose.....	14
GUI ON-OFF.....	14
HMD repose.....	14
Desktops <-> VR View.....	14
Level Update (tilted view).....	14
Level Zero (front view).....	14
Shading:.....	14
Render engine:.....	14
Render settings:.....	15
4Kx4K for EQD, REL or VR180.....	15
8Kx4K for VR360.....	15
Miscellaneous:.....	15
Hide Sidebar.....	15
Reframe Camera Bounds.....	15

Use Scene Camera.....	15
Other:.....	15
The workflow process for VR art creation with VHT tools.....	16
How to view the rendered images or videos in VR.....	18
Visual Quality and Performance Evaluation.....	18
1 st Art preview in VR.....	19
2 nd Rendering the final VR art.....	19
3 rd Consuming the VR art.....	19
Demo samples.....	19
Acronyms.....	20
Links.....	21
Revision History.....	21

Introduction

“VHT tools for Blender” is a Python script add-on for Blender 3D modeling software specially designed as a concept (lets call it “Spatial Modeling”) to help CG artists to create 3D stereoscopic 180 or 360 degree panoramic computer generated art images or videos for VR users.

These tools will create a new window to draw a stereoscopic Sbs 180 or 360 degrees projection for live scene view in VR and will help artist to render stereoscopic VR images or videos files.

To view in VR the live stereoscopic viewport, will require to use the Virtual Home Theater (VHT) VR application from Steam store and to have two displays: one for Blender GUI and the second for stereoscopic viewport!

The render tools can be used also without the Virtual Home Theater (VHT) VR application from Steam store or dual displays!

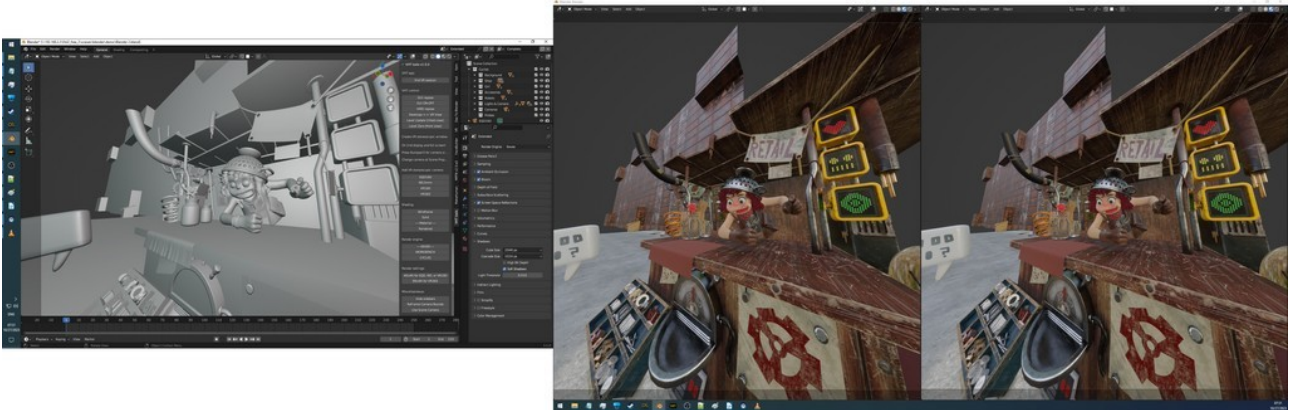
When using these tools, the artist will work as usual in front of his computer display and will use the VR headset and VHT when a VR preview is needed, in a sited position using his mouse, keyboard and a SpaceMouse.

In VR the artist will see the main desktop screen where will be his Blender GUI main window and surrounded by his ‘true to scale’ 3D scene, rendered live as stereoscopic 3D 180 or 360 degree panorama using any real-time render engine available from Blender: Cycle (path-tracer), Eevee, etc.

VHT tools can be used by a VR CG artist:

- for live stereoscopic scene preview in VR before final rendering
- to render final stereoscopic images or videos for publishing
- to check objects geometry for proportions and dimensions
- to check for 3D disparity and select the best IPD for stereoscopic cameras
- for 3D modeling, morphing and posing
- to check scene composition and animations
- for lighting setup, colors, materials, texture, environment and tone mapping tweaking
- to find the best scene view for VR (camera angle)
- to select the best VR lens projection/mapping

Concept Preview



This image is a desktop snapshot demo sample of what a user will see on his dual displays.



This image is a VR snapshot demo sample of what a user will see through the HMD lens.

Scene source:

<https://cloud.blender.org/p/gallery/5dd6d7044441651fa3dec56>

The same concept product with additional extra features is also available from VHT for Daz Studio 3D modeling software.

Use Case Examples

An artist can use these tools to create images or videos in stereoscopic 3D format for VR that are more immersive and spectacular than a classic presentation on a 2D flat surface (paper or screen) for:

- Architectural interior/exterior design presentation
- Vehicle (car, airplane, yacht etc.) interior/exterior design presentation
- Cloth design on a character for fashion presentation
- Furniture design presentation
- Engineering design evaluation
- Epic VR dioramas of historic events
- Visual novels for VR users
- re-rendering of old 3D scenes for VR users
- VR movies creation
- a game or movie character showcase

Software Requirements

- Windows 10
- **Blender** 3.x (for 3D modeling and rendering)
- **VHT tools for Blender**
- **VHT** 1.18.0 (only for live 3D stereoscopic scene preview from Blender and to view the final rendered images or videos)
- **FFmpeg** (used by VHT render tool script to create videos from the rendered animated frames)

Go to:

blender.org

virtualhometheater.com

ffmpeg.org

Hardware Requirements

The following hardware is required only when using VHT VR app for VR preview!

- a **2nd display** (real, dummy or virtual; **2K** or **4K** native/custom) for the VR stereoscopic Viewport window
- **VR HMD** (high PPD PC models like **HP G2** or better for clarity)
- **NVIDIA RTX GPU** (the bigger VRAM the better, **4090** for performance and fast renders)
- **SpaceMouse** for 3D scene navigation

Remarks:

- **If you have only one display you will not be able to start VHT in Spatial Modeling mode.**
- The VR controllers are not required and should not be used.
- The SpaceMouse is required for ease of navigation through the scene and can be ignored if you use other preferred input methods.
- The 2nd display resolution will define the image clarity (visual PPD) for the live 3D stereoscopic scene preview in VR.

Optional: **headphones** for binaural music listening from VHT when creating art.

Highlighted Features

- 4 stereoscopic camera types
- 3 lens projection/mapping models
- render final VR images or videos
- tool to control the VHT virtual displays
- start VHT app and create an IPC between VHT app and 'VHT tools for Blender' script

Installation

Unzip the "VHT tools for Blender.zip" file.

In Blender go to "Edit > Preferences.. > Add-ons > Install..." then select *VHT_tools_for_blender.py* to install.

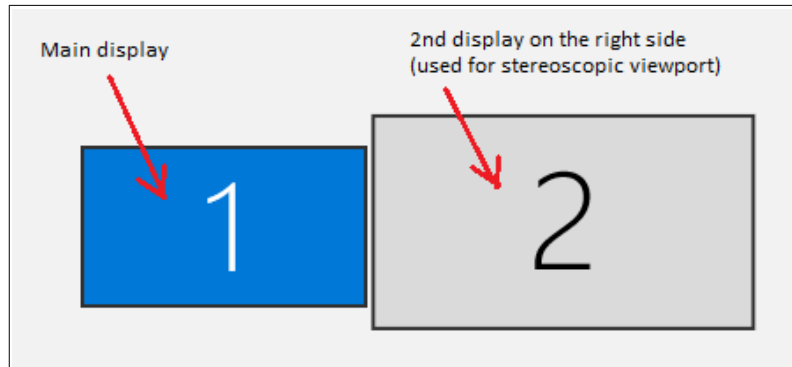
Filter the add-on by keyword "vht" then **Enable** the VHT add-on.

The add-on location is "**View3D > Sidebar > VHT tools**".

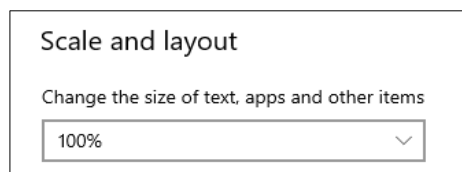
Prerequisite Settings

Set the display with the smallest resolution as the main display (recommended).

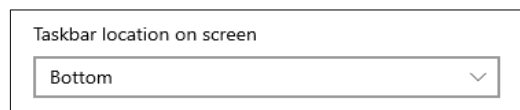
Rearrange your displays like shown below:



Set the scale for the 2nd display to 100% (very important requirement)



For the 2nd display set the taskbar location on screen to **Bottom** (very important requirement)

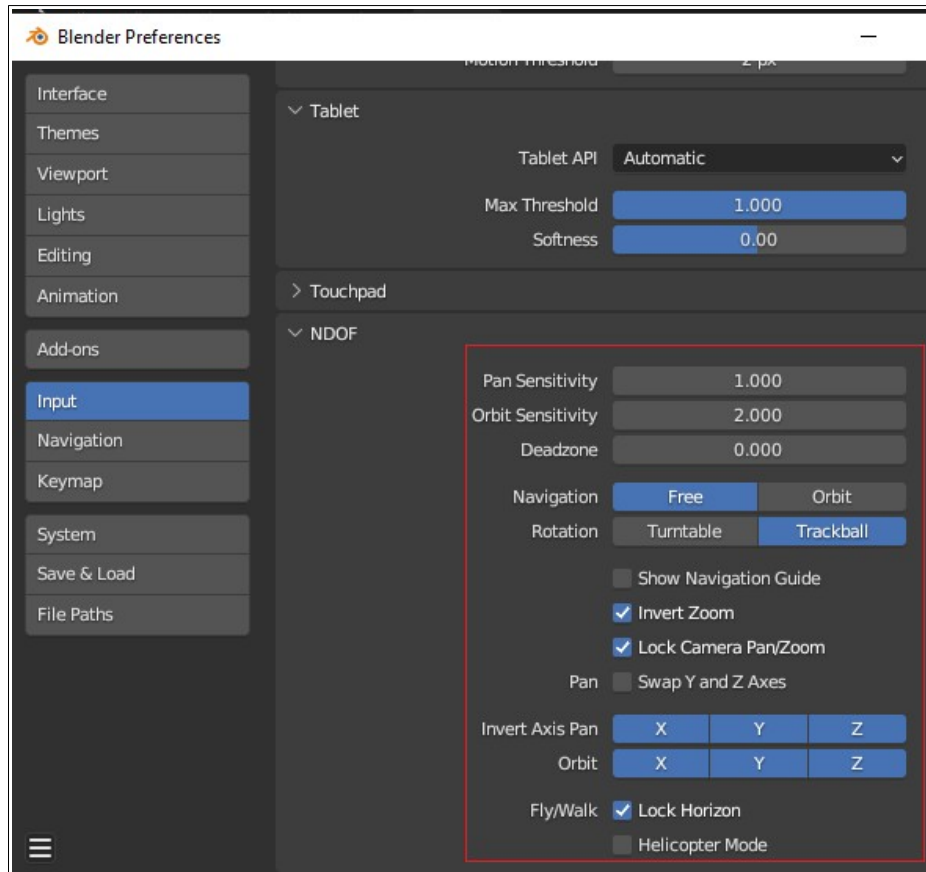


If the second display has a lower native resolution than 2K (e.g. 1920x1080) you can try to change resolution by using the “[NVIDIA Control Panel > Display > Change resolution > Customize.. > Create Custom Resolution...](#)” type 2560x1440 or 3840x2160, then chose the new resolution from Display Settings.

Blender Preparation

The main window from Blender app must be placed on the main display.

For SpaceMouse the recommended settings are:



Quick First Use

Assuming you have done the prerequisite preparations as described above, do the next steps to start a VHT-Blender VR work session:

1. Log-in to your Steam account.
2. Turn On your VR HMD.
3. In Blender load the scene you want to see/work in VR.
4. Click on “VHT_tools > Create VR stereoscopic window: **EQD180 SbS crosseye**”
5. Move the newly created SbS window to the second display and maximize it (<https://youtu.be/kZCJ0YcLq8o>).
6. **Start VHT from Blender (not from Steam library)** by clicking on “VHT_tools > VHT app: **Launch VR session**”
7. Put your VR HMD on your head.
8. If necessary use your mouse to bring Blender windows to front. Use the mouse and keyboard to interact with Blender as before but in VR.
9. If necessary press Numpad 0 to activate scene camera view.
10. If necessary use “VHT_tools > VHT control: **HMD repose**” and/or “VHT_tools > VHT control: **GUI repose**” to recenter HMD and GUI in VR.
11. Use your **SpaceMouse** to navigate through scene. If you don't have a SpaceMouse read the remarks from Required Hardware chapter above.
12. You can change the render engine for VR to “VHT_tools > Render engine: **CYCLES**”

VHT Tools Interface Description

VHT tools add-on is shown on the sidebar of a viewport if an active camera is selected for the scene.



Cameras

All VR cameras are stereoscopic with an IPD of 65 mm as default.

A match between the human eye IPD and the camera IPD is not necessary (use what is best for your scene context):

- Use a lower IPD than 65 mm for close objects or 360 panoramic views.
- Use very large IPD (meters) if you want to have a toy/miniature-like view renders.

For view rotation and translation press Numpad 0 to activate scene camera view.

VR180 and VR360 cameras use the synthetic stereoscopic panoramic rendering technique described here <https://code.blender.org/2015/03/1451/>

Add VR stereoscopic camera:

EQD180

Add a stereoscopic camera with an Equidistant [1] projection/mapping lens.

IPD = 65 mm, FoV = 180 degree, valid only for CYCLE render engine.

REL9mm

Add a stereoscopic camera with a Rectilinear projection/mapping lens.

IPD = 65 mm, focal length = 9 mm, FoV = 127 degree, valid for any render engine.

VR180

Add a stereoscopic camera with a Spherical Equirectangular Panorama [1,2] projection/mapping lens. IPD = 65 mm, FoV = 180 degree, valid only for CYCLE render engine.

VR360

Add a stereoscopic camera with a Spherical Equirectangular Panorama [1,2] projection/mapping lens. IPD = 65 mm, FoV = 360 degree, valid only for CYCLE render engine.

[1] If other render engine than CYCLES will be used for EQD180, VR180 or VR360 cameras then a 9 mm focal (127 degree FoV) rectilinear projection/mapping lens will be used for preview. Only CYCLE render engine can render Spherical Equirectangular Panoramas or EQD projection/mapping.

[2] For VR180 and VR360 cameras the Spherical Equirectangular Panorama projection/mapping is not active (Blender bug!?) when you view the viewport live in VR resulting in 3D disparity especially for VR360 when watching behind. Fortunately when you do actual final render the image is rendered correctly using the Spherical Equirectangular Panorama projection/mapping resulting in no 3D disparity.

Create VR stereoscopic window:

Use the following options when you create a new scene or load a scene without a VR camera or stereoscopic window.

EQD180 SbS crosseye

Create a SbS split layout new viewport window and add one EQD180 camera.

VR180 SbS crosseye

Create a SbS split layout new viewport window and add one VR180 camera.

EQD360 SbS crosseye

Create a SbS split layout new viewport window and add one VR360 camera.

Blender to VHT messaging bridge:

To view the stereoscopic Sbs Viewport in VR it is necessary to start VHT VR app in desktop mirror spatial modeling mode using “*Launch VR session*” tool script to create a messaging bridge between Blender and VHT.

This bridge will make VHT know when drawing style and image projection/mapping are changing on Blender Viewport to change the used VHT projection accordingly.

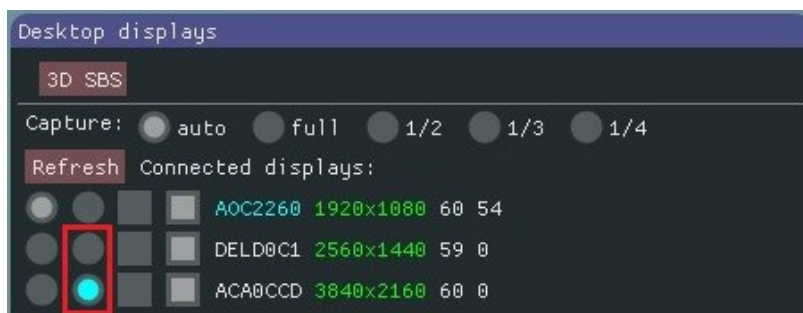
The messaging bridge will be used also to send user requested commands to VHT from Blender using the following tool scripts:

- GUI_repose
- GUI ON-OFF
- HMD repose
- Desktops <-> VR View
- Level Update (tilted view)
- Level Zero (front view)

VHT app will be used by user to view in VR the stereoscopic Sbs Viewport as a surround scene environment with an overlay main desktop display (Blender GUI).

When VHT is started it will chose automatically the highest resolution display as the source of the stereoscopic Viewport image.

You can change the source of stereoscopic image to be another display (if you have more than two) by using the VHT UI:



Launch VR session

This will launch VHT app in desktop mirror spatial modeling mode and will create a messaging bridge between Blender and VHT.

End VR session

This will end Blender to VHT messaging bridge and close VHT app.

Rendering

Before rendering it is recommended to end any VR app that is using GPU resources required for rendering.

These rendered images are using the cross-eye format so you can preview them with your own eyes without any VR device using the cross-eye method.

This render script will use the scene name for saving the rendered images or videos using the following format:

SceneName R_Date-Time_CameraName_Format_TiltAngle_.FileType

Example: “MyScene R_2023-10-26-12-54-7_EQD180_FF3D_SbS_crosseye_tilt-9_.tif”

The rendered images and videos are saved to the scene folder.

To view the results (images and videos) use the built-in image viewer or video player from VHT VR app.

When start to render the VHT tools script will try to close the running VR applications to free GPU resources.

Start to render IMAGE:

Will start Blender render for current scene camera.

Quick at 16 samples + den

will use: Resolution = full, Samples = 16, Denoise = On

at 1024 samples + den

will use: Resolution = full, Samples = 1024, Denoise = On

as SET

will use the current user defined render settings

Start to render VIDEO:

Will start to render video frames using Blender render for current scene camera.

This render script will not use the built-in video render from Blender but instead will execute an ffmpeg.exe command to create a video from rendered frames.

The FFmpeg input arguments used to encode a video are:

- when using alpha transparency: `-c:v vp9 -b:v 0 -crf 23 -pix_fmt yuva420p`
- for non-transparent video: `-c:v hevc_nvenc -b:v 0 -crf 23`

To create a transparent video you need to set the file format to PNG and the color mode to RGBA from “Blender > Output properties > Output”.

For FFmpeg commands to work you need to add the FFmpeg folder path to “Windows_10 > System Properties > Advanced > Environment Variable > User variables for admin > Path > Edit > New”.

Quick at 25% + 16 sam + den

will use: Resolution = ¼ of X & Y resolution, Samples = 16, Denoise = On

at 1024 samples + den

will use: Resolution = full, Samples = 1024, Denoise = On

as SET

will use the current user defined render settings

VHT Control:

These control scripts will work only if Blender to VHT messaging bridge is created by launching VHT VR app with “*Launch VR session*” tool button from Blender interface.

GUI_repose

Repose the floating virtual desktop screen in front of you or on another location.

GUI ON-OFF

Turn Off/On the floating virtual desktop screen.

HMD repose

Repose and recenter HMD in VR space.

Desktops <-> VR View

Switch between VR view and desktop view mode for screens.

VR view (default spatial modeling mode) will display the stereoscopic Viewport from Blender as a 3D sphere projection/mapping.

Desktop view will show all desktops as flat screens in front of the user.

Level Update (tilted view)

Level the scene view in VR as the Viewport camera tilt angle from that moment.

Use this to preview how the final render image will be displayed in VHT when the built-in image viewer will be used.

Level Zero (front view)

Show the scene view in VR at zero degree tilt angle (default).

Shading:

Set the shading you want for stereoscopic Viewport used for VR live preview.

Render engine:

Set the render engine you want for stereoscopic Viewport used for VR live preview.

Render settings:

4Kx4K for EQD, REL or VR180

Use this setting for Side-by-Side output rendering format file type.

8Kx4K for VR360

Use this setting for Top-and-Below output rendering format file type (only for 360 panorama).

Miscellaneous:

These are done automatically by VHT tools script when working with VHT tools.

Use them if necessary.

Hide Sidebar

Will hide sidebars (if visible) from the left and right frames on VR viewport window.

Reframe Camera Bounds

Will reframe camera bounds from the left and right frames on VR viewport window.

Use Scene Camera

Use the scene camera for VR viewport window.

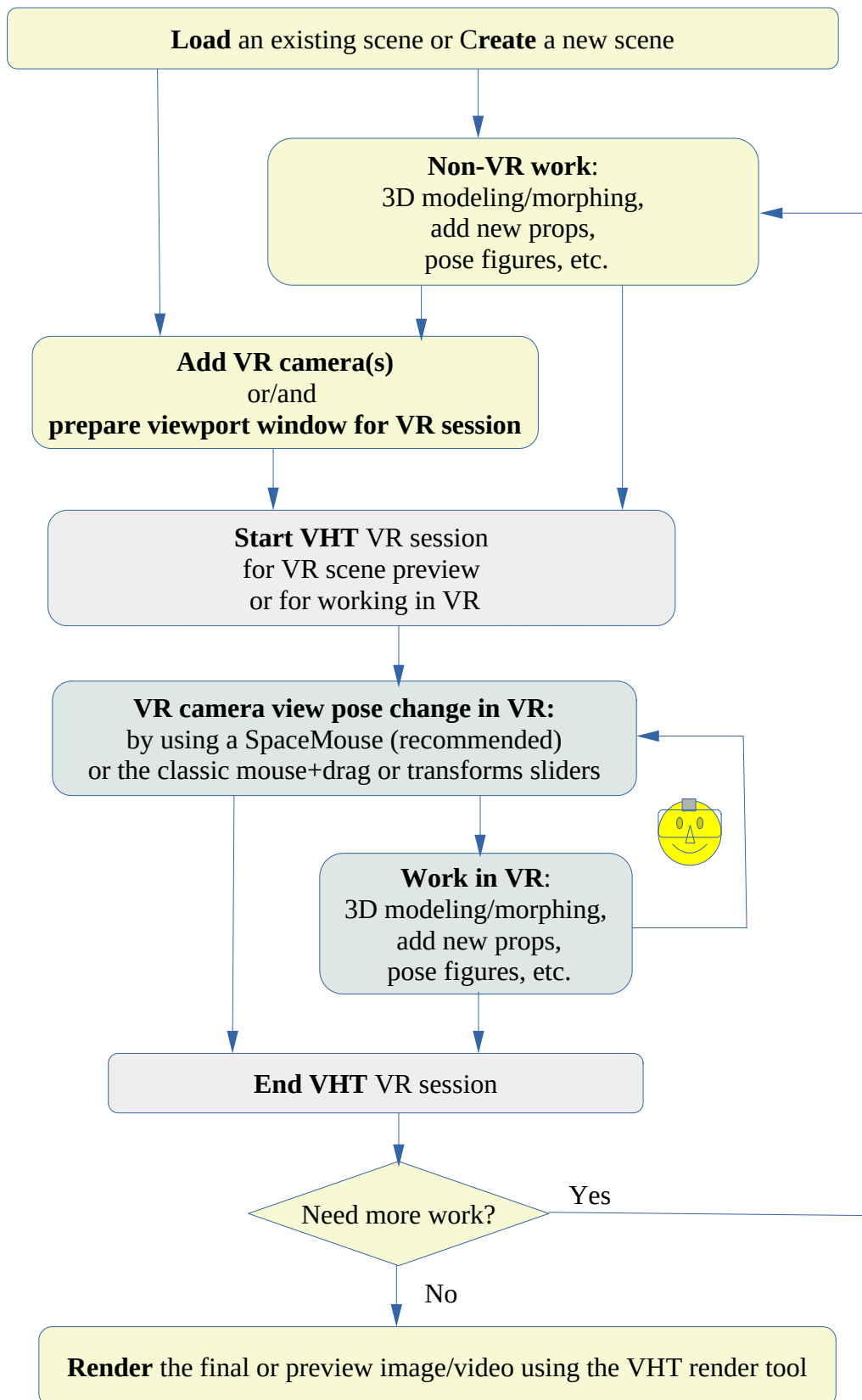
Other:

When using SOLID shading the following settings will show: Light, Color type, Show shadows, Cavity type, Show object outline and Show specular highlight.

Use these to tweak the drawing style when preview the scene in VR.

The workflow process for VR art creation with VHT tools

#1 This is the case when you have the VHT VR app (demo or paid version) installed AND you also have two displays.

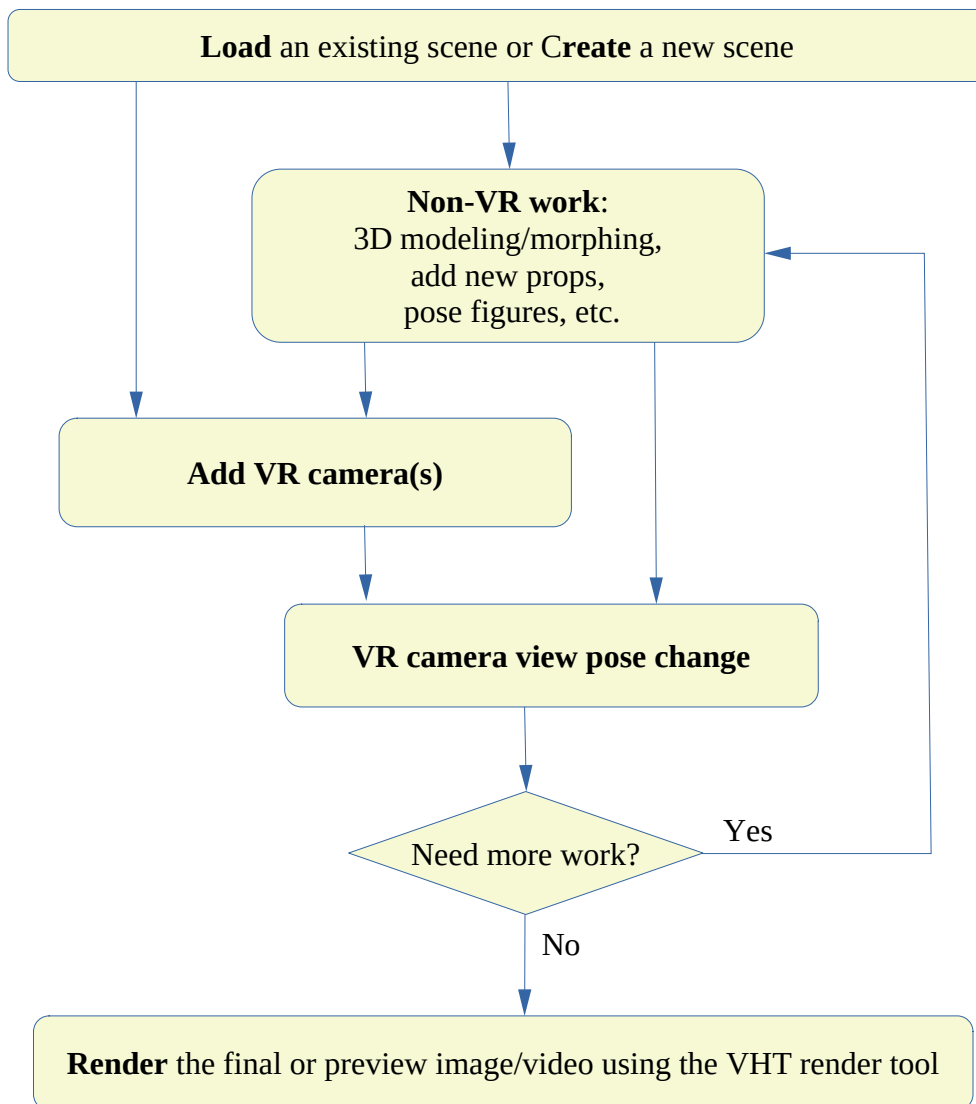


#2 The next case is when you don't have the VHT VR app (demo or paid version) installed OR you don't have two displays.

Some tools (VR cameras and render tool) can be used without VHT VR app but other tools (VR live scene preview in VR) will require VHT VR app.

When using the "VHT tools for Blender" with VHT VR app is like using a photographic camera with a viewfinder and a tripod.

When using the "VHT tools for Blender" without VHT VR app is like using a photographic camera and a tripod but without the viewfinder.



How to view the rendered images or videos in VR

Use the built-in image viewer or video player from VHT (recommended) or use other VR software.

Method 1:

Right click on the image or video then “Open with” and choose VHT.

Method 2:

Use the VHT’s built-in file explorer to browse to the file location then click View or Play.

Method 3:

Use other VR software capable to correctly view the rendered stereoscopic images or videos.

Do not remove or rename the format keywords from the media file names.

For example:

Demo R_2023-10-26-12-54-7_EQD180_FF3D_SbS_crosseye_tilt-9_.png

you can rename the “*Demo R_2023-10-26-12-54-7*” text but keep the “*_EQD180_FF3D_SbS_crosseye_tilt-9_*” format keywords!

Visual Quality and Performance Evaluation

We will evaluate the quality of stereoscopic panoramic images viewed through the HMD using the PPD property.

The type of PPD can be:

- **biological:** the human eye can have a 60 PPD in the center foveal region for 20/20 vision
- **hardware:** a consumer grade HMD can have today a 24 PPD (HP G2)
- **format:** image file resolution divided by the projection angle (e.g. for 4096x4096 at 180 degree the $PPD = 4096/180 = 23$)
- **software:** this is lower or equal with the format PPD and depends on the compression level, render samples and denoise level

The quality of an image in VR will be defined by the software PPD and limited by format and hardware PPD.

The perceived PPD (what you actually see in VR) will be the minimum PPD of the above types.

For the following use cases (art creation and consuming art) we will use a HP G2 and NVIDIA RTX 4090 GPU.

1st Art preview in VR

This is the case when an artist use VHT tools for Blender to create VR art in VR for VR users.

He has a 2nd display with a resolution of 2560x1440 @ 60Hz.

When using the Eevee or OpenGL raster render engine for EQD180 or REL9mm the software PPD will be always equal with the format PPD of $2560/2/127 = 10$ (close to Valve Index HMD PPD of 13) at a variable and lower than 60 fps depending on the complexity of the scene (usually will be at 60 fps for most cases).

When using Cycles render engine with 16 samples for Viewport and denoiser for EQD180 the software PPD will be lower than theoretical format PPD of $2560/2/180 = 7$.

For each render cycle the software PPD will increase from zero to 7 PPD in 16 steps at a rate of 2 fps. Depending on the scene complexity the path traced image will be updated at a rate of <2 fps (this fps is OK for a static scene turn based live preview ray-traced render in VR).

2nd Rendering the final VR art

This is the case when an artist is happy with the scene he created and want to share his art with others. He will use the VHT tools for Blender to render the VR art.

You should expect double render duration for a stereoscopic image due to the format nature of this kind of image (need one image for left and another for right eye) for a given resolution.

If you choose the best render setting: >1024 samples, png or tif, and a color depth of 16 bits you should expect the software and format PPD for:

- EQD180 at 4096x4096/eye resolution: $4096/180 = 23$ PPD
- REL9mm at 4096x4096/eye resolution: $4096/127 = 32$ PPD
- VR360 at 8192x4096/eye resolution: $8192/360 = 23$ PPD

Render times will range from minutes to few hours per one final stereoscopic (left+right) image depending on scene complexity, format and software target PPD, GPU, etc.

3rd Consuming the VR art

This is the case when a VR user will admire your VR art.

For a rendered EQD180 image he can enjoy a $4096/180 = 23$ PPD image details which is a perfect match to his HP G2 VR HMD of 24 PPD.

He will be very impressed by this 4K per eye 3D VR panormaic image.

Demo samples

For rendered images go to <https://virtualhometheater.com/>

Acronyms

- VHT – Virtual Home Theater
- HMD – Head Mounted Display (VR device)
- CGI – Computer Generated Image
- CG – Computer Graphics
- IPD – Interpupillary distance (distance between two cameras or eyes)
- PPD – pixels (details) per angle degree (hardware or visual)
- SbS – Side-by-Syde (screens layout format for stereoscopic images)
- REL – Rectilinear (photographic projection/mapping)
- EQD – Equidistant (photographic projection/mapping)
- FoV – Field of View
- GUI – Graphical User Interface
- UI – User Interface
- PC – personal computer
- FPS – frames per second (image refresh rate)
- IPC – inter-process communication (messaging bridge)

Links

Virtual Home Theater:

- <https://virtualhometheater.com/>
- <https://store.steampowered.com/app/989060/>
- <https://www.youtube.com/@VHTtools>
- <https://www.youtube.com/@virtualhometheater3887>
- https://youtu.be/ihrxFmL-A_Y
- <https://youtu.be/h2ejyZLwvtM>

Blender:

- <https://www.blender.org>
- <https://www.blender.org/download/demo-files/>
- <https://docs.blender.org/manual/en/latest/render/eevee/index.html>
- <https://docs.blender.org/manual/en/latest/render/cycles/index.html>
- <https://www.cycles-renderer.org/>

Spherical Stereoscopic Panoramas:

- <https://code.blender.org/2015/03/1451/>
- <https://paulbourke.net/papers/vsimm2006/>

<https://ffmpeg.org/>

Revision History

- v1.0 – Feb 17, 2024: First release