SGO

# mistika VR

USER MANUAL

SGO


# Introduction

# 1 Introduction

Mistika VR is an affordable Virtual Reality-focused software solution for optical flow stitching that, in just one click, achieves both quicker than real-time speeds for 4K VR media encoding and stabilizes 360º footage. Mistika VR has revolutionized multi-camera stitching, reducing the time needed to correct footage from hours down to minutes – and all at the highest possible quality.

Many VR camera manufacturers have adopted Mistika VR as their first choice solution for providing metadata that ensures a faster, better and more precise stitch.

Other Mistika VR features include a Stereo 3D toolset, enabling the creation of truly immersive content, a one-click Stabilization process which easily and quickly smoothes out shaky footage, and Keyframe Animation, which provides both enhanced flexibility for stitching and greater control of the VR 360º post-production process.

*This version is updated to Mistika VR 8.11*

SGO


# Installation and configuration

# 2 Installation and configuration

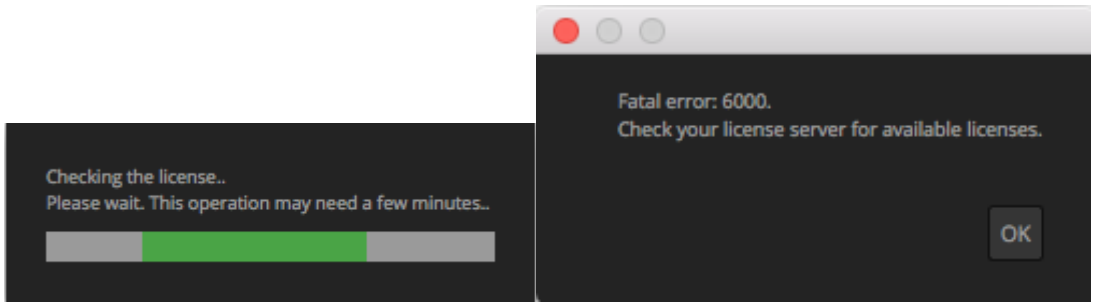
First, create your account at SGO (remember to confirm your account by clicking the link you will receive in your email).

Choose your installation plan at SGO VR Plans and then download the software installer, run it and follow the instructions.

## 2.1 Activating a Mistika VR license with Internet Connection

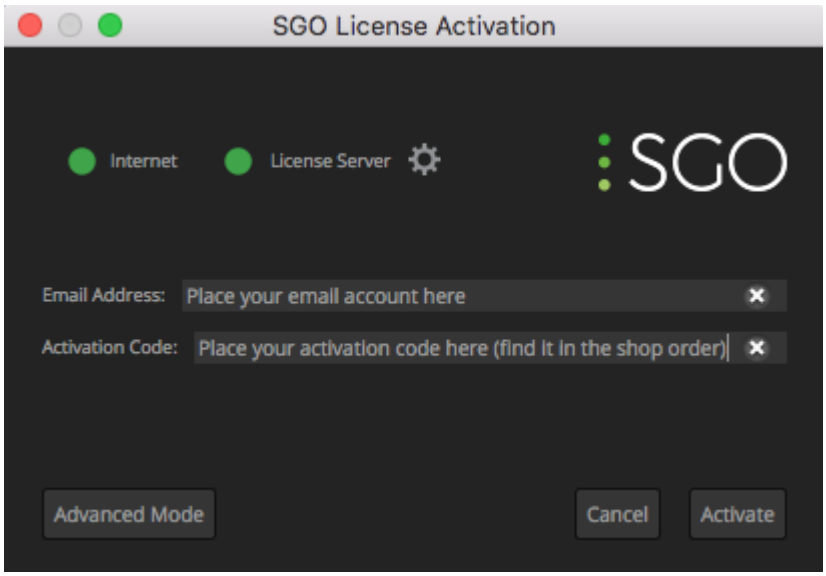
Activation codes and license management are processed either by means of the SGO License Activation Tool or directly in your SGO web account (SGO My account). We recommend you use the SGO License Activation Tool, as this installs your licenses automatically, and also provides more information if problems arise. Your activation codes can also be managed through your SGO web account, if, for example, you need to manage theses codes for other users within your company and do not have the SGO License Activation Tool in your device.

Once installed, open Mistika VR. You will see the following messages:




The SGO License Activation Tool will then open when you must your Activation Code. You can find your Activations Code in the Shop Order sent to the email address provided at the time of purchase; you can also find it in your SGO Account (SGO Activation Codes).

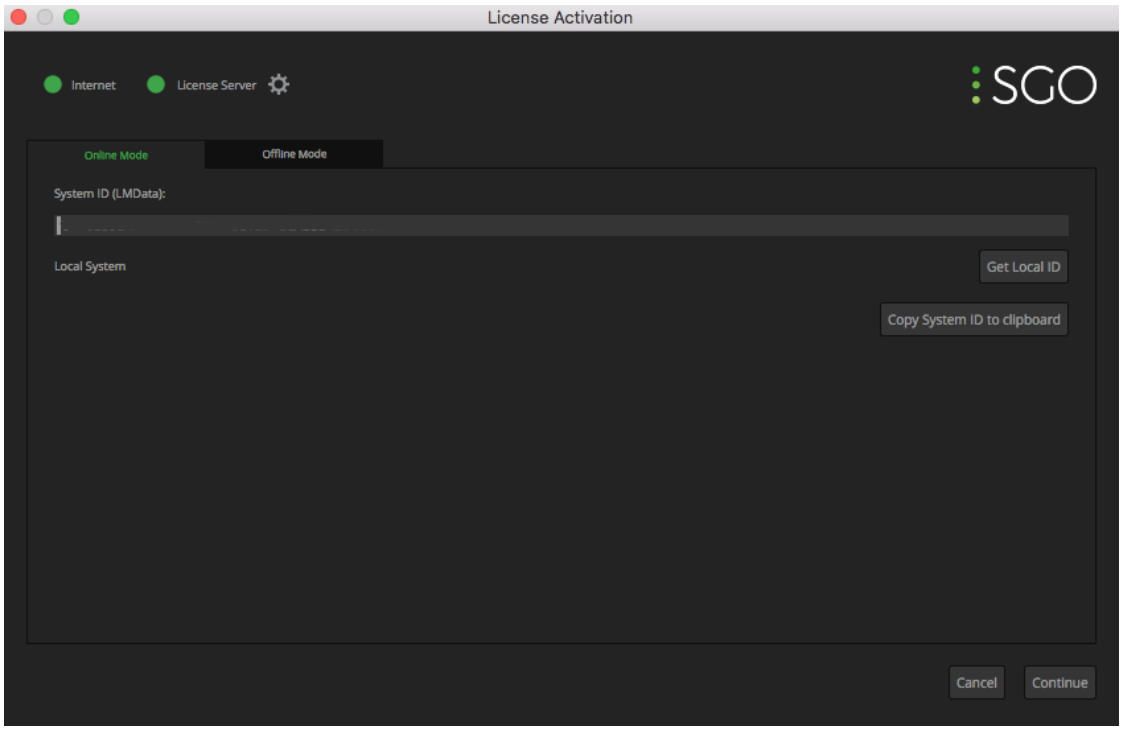
By default, the single mode License Activation window will open:




## 2.2 Activating a Mistika VR license without Internet Connection

If you want to activate a Mistika Technology product on a computer not connected to the internet, follow these steps:

- Get the LMData identifier from the computer on which the Mistika Technology product is to be used (no internet connection is required):



- Type the LMData code into a computer connected to the Internet, and use the SGO Activation Tool to get a License file for it.
- Transfer the License file to the original computer and install it with the SGO Activation Tool.

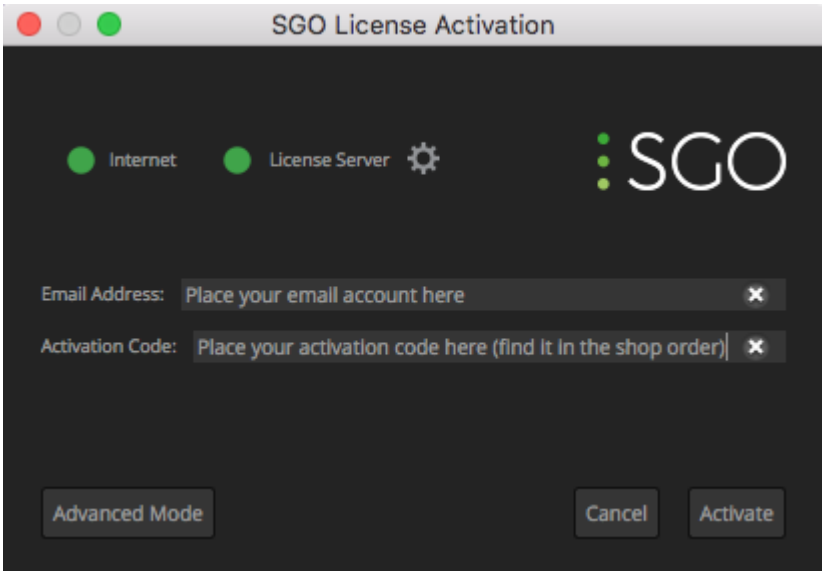
**Complete procedure for the Offline method:**

1. Using any computer with Internet connection, purchase an Activation Code at SGO, and download the Mistika software.

2. The computer to be activated (on which the Mistika software is to be used) does not need Internet connection. Simply install the software acquired in the previous step and run the SGO Activation Tool. A prompt warning about the lack of Internet connection and License Server will appear, which you can ignore. Copy the LMData line that appears in the *System ID* field.

3. Access a computer with Internet connection and open the SGO Activation Tool (this computer must also have Mistika software installed; although Mistika will not be used on this computer, it needs the SGO Activation Tool that comes with the software).

4. Select *Advanced* mode.




5 . Input the LMData code that you copied in step 2, and press *Continue*.

6. Click *Add Activation Code*, then input your Activation Code and press *Validate & Activate*.

7. Once the activation is completed, it will provide a License file for the original system.

8. Transfer the license to the original system and open *SGO License Activation Tool>Offline*. Input the license that you acquired in the previous steps.

*NOTE: There is also another way to get licenses without Internet connection, which is to use floating licenses. In this method, all the activation codes are installed in only one computer, which will be the 'License Server', and all client computers will obtain the licenses from it. The client computers will not need Internet connection, only a connection to the License Server. At the same time, the license server can activate the codes either with an internet connection or by using the offline method explained above (requiring identical steps, but using the LMData of the license server).*

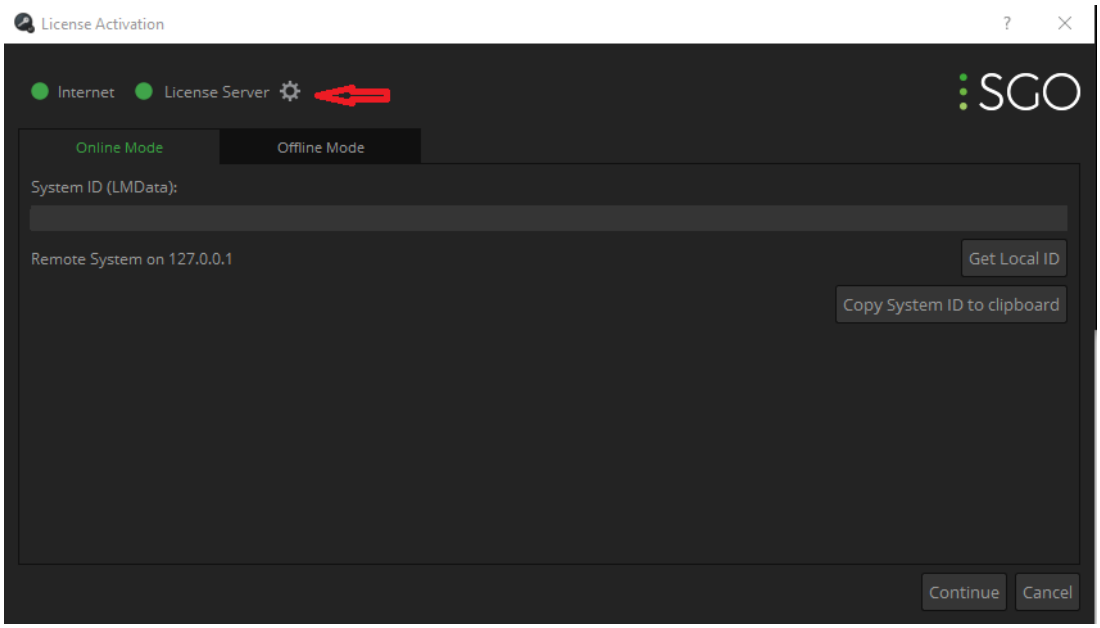
## 2.3 How to configure a floating license in Mistika VR

To configure one system to read the license from a remote License Server, follow this simple steps:

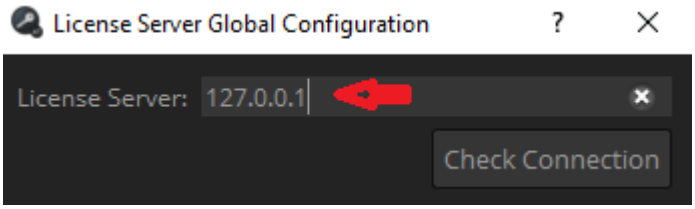
Nothing special needs to be done in the server computer: just install the Mistika software and activate all your activation codes on it. All Mistika licenses are floating licenses, so they can be used across systems (as long as there are no other clients connected at the same time than those with activated codes).

Now, install the Mistika software in the client computers, and do the following:

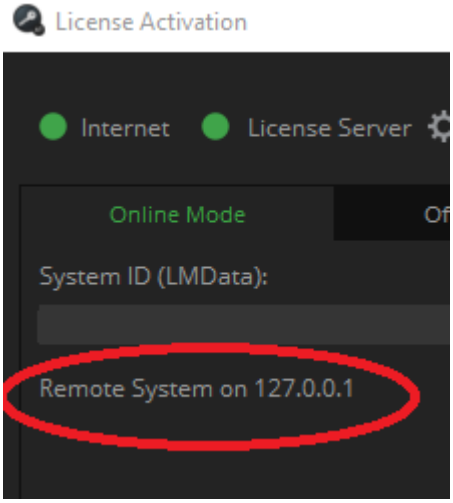
1. Open the **SGO License Activation Tool** and enter the server config dialog:




2. A dialog box will appear, specify the desired License Server IP:

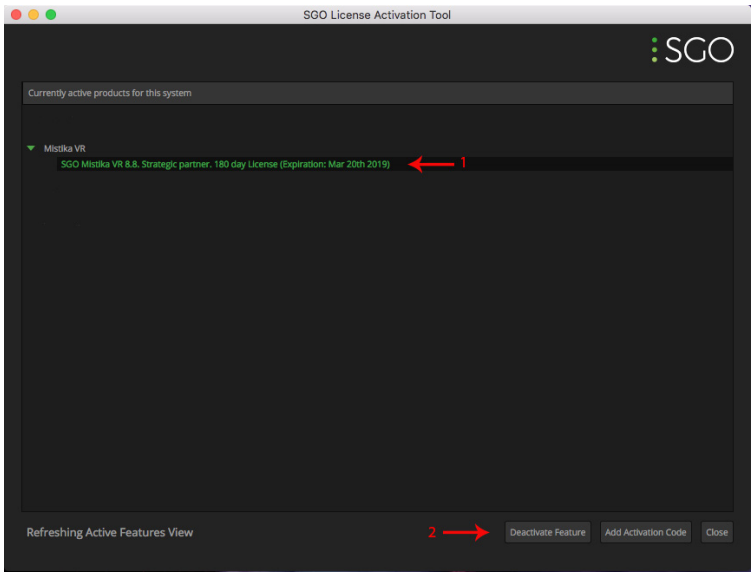



3. If the software can connect to the remote server, the License Server information will be displayed in the mail activation window GUI, as follows:



## 2.4 Active licenses

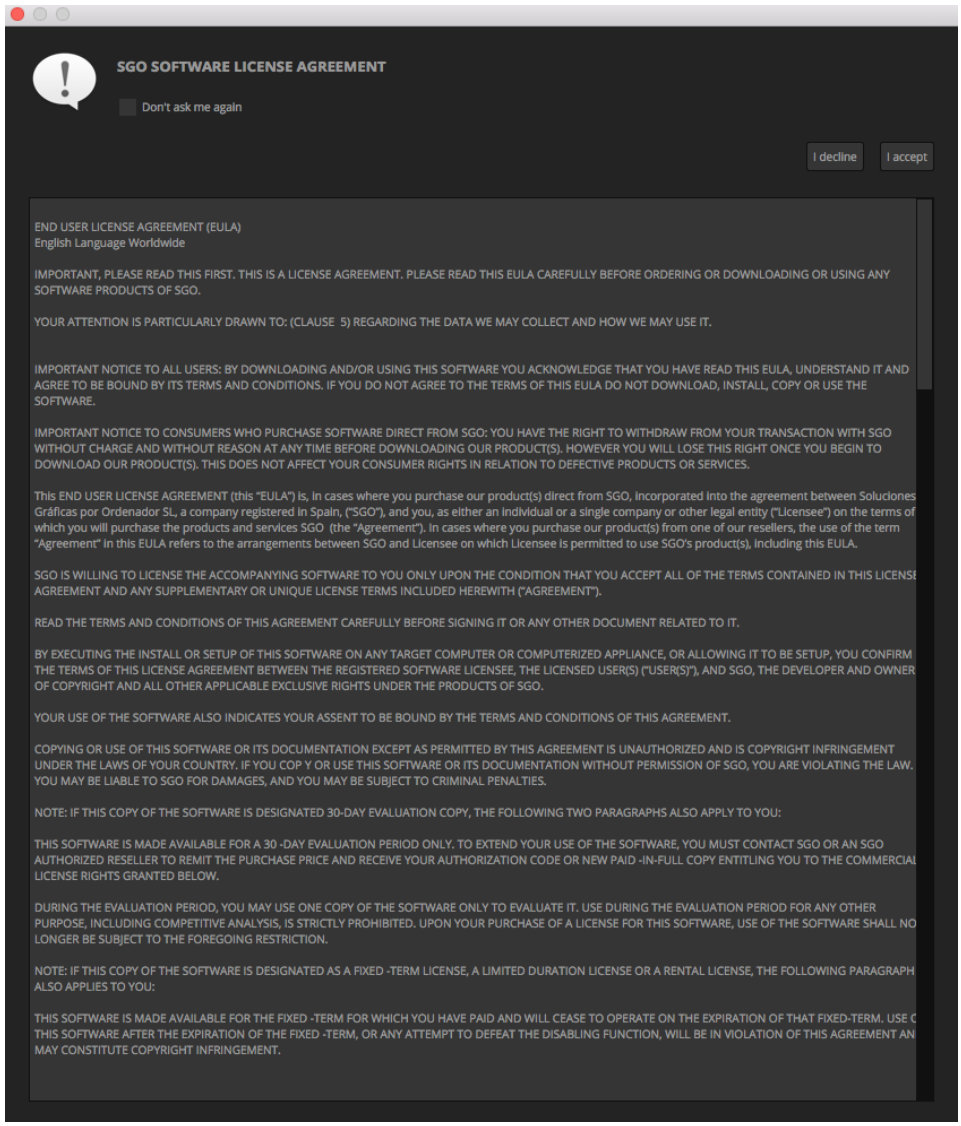
All active licenses linked to your system can be found in the Advanced mode of the SGO License Activation Tool. This allows the deactivation of licenses by double-clicking on *Deactivate Feature*, as follows:




Please note that deactivating a license key will not cancel your subscription. All subscriptions must be managed through SGO Subscriptions.

## 2.5 First launch of Mistika VR

On the first launch of Mistika VR, an **SGO Software License Agreement** prompt will appear. To read the whole license agreement, use the scroll-bar to the right. To access Mistika VR, you must Accept the Software License Agreement. If you don't want to see this prompt again, click *Don't ask me again* at the top of the window. Once the License Agreement is accepted, Mistika VR will launch. On its first launch, Mistika VR will load the factory default window layout.


SGO SOFTWARE LICENSE AGREEMENT
Don't ask me again
I decline
I accept
END USER LICENSE AGREEMENT (EULA)
English Language Worldwide
IMPORTANT, PLEASE READ THIS FIRST. THIS IS A LICENSE AGREEMENT. PLEASE READ THIS EULA CAREFULLY BEFORE ORDERING OR DOWNLOADING OR USING ANY SOFTWARE PRODUCTS OF SGO.
YOUR ATTENTION IS PARTICULARLY DRAWN TO: (CLAUSE 5) REGARDING THE DATA WE MAY COLLECT AND HOW WE MAY USE IT.
IMPORTANT NOTICE TO ALL USERS: BY DOWNLOADING AND/OR USING THIS SOFTWARE YOU ACKNOWLEDGE THAT YOU HAVE READ THIS EULA, UNDERSTAND IT AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA DO NOT DOWNLOAD, INSTALL, COPY OR USE THE SOFTWARE.
IMPORTANT NOTICE TO CONSUMERS WHO PURCHASE SOFTWARE DIRECT FROM SGO: YOU HAVE THE RIGHT TO WITHDRAW FROM YOUR TRANSACTION WITH SGO WITHOUT CHARGE AND WITHOUT REASON AT ANY TIME BEFORE DOWNLOADING OUR PRODUCT(S). HOWEVER YOU WILL LOSE THIS RIGHT ONCE YOU BEGIN TO DOWNLOAD OUR PRODUCT(S). THIS DOES NOT AFFECT YOUR CONSUMER RIGHTS IN RELATION TO DEFECTIVE PRODUCTS OR SERVICES.
This END USER LICENSE AGREEMENT (this "EULA") is, in cases where you purchase our product(s) direct from SGO, incorporated into the agreement between Soluciones Gráficas por Ordenador SL, a company registered in Spain, ("SGO"), and you, as either an individual or a single company or other legal entity ("Licensee") on the terms of which you will purchase the products and services SGO (the "Agreement"). In cases where you purchase our product(s) from one of our resellers, the use of the term "Agreement" in this EULA refers to the arrangements between SGO and Licensee on which Licensee is permitted to use SGO's product(s), including this EULA.
SGO IS WILLING TO LICENSE THE ACCOMPANYING SOFTWARE TO YOU ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS LICENSE AGREEMENT AND ANY SUPPLEMENTARY OR UNIQUE LICENSE TERMS INCLUDED HEREWITH ("AGREEMENT").
READ THE TERMS AND CONDITIONS OF THIS AGREEMENT CAREFULLY BEFORE SIGNING IT OR ANY OTHER DOCUMENT RELATED TO IT.
BY EXECUTING THE INSTALL OR SETUP OF THIS SOFTWARE ON ANY TARGET COMPUTER OR COMPUTERIZED APPLIANCE, OR ALLOWING IT TO BE SETUP, YOU CONFIRM THE TERMS OF THIS LICENSE AGREEMENT BETWEEN THE REGISTERED SOFTWARE LICENSEE, THE LICENSED USER(S) ("USER(S)"), AND SGO, THE DEVELOPER AND OWNER OF COPYRIGHT AND ALL OTHER APPLICABLE EXCLUSIVE RIGHTS UNDER THE PRODUCTS OF SGO.
YOUR USE OF THE SOFTWARE ALSO INDICATES YOUR ASSENT TO BE BOUND BY THE TERMS AND CONDITIONS OF THIS AGREEMENT.
COPYING OR USE OF THIS SOFTWARE OR ITS DOCUMENTATION EXCEPT AS PERMITTED BY THIS AGREEMENT IS UNAUTHORIZED AND IS COPYRIGHT INFRINGEMENT UNDER THE LAWS OF YOUR COUNTRY. IF YOU COP Y OR USE THIS SOFTWARE OR ITS DOCUMENTATION WITHOUT PERMISSION OF SGO, YOU ARE VIOLATING THE LAW. YOU MAY BE LIABLE TO SGO FOR DAMAGES, AND YOU MAY BE SUBJECT TO CRIMINAL PENALTIES.
NOTE: IF THIS COPY OF THE SOFTWARE IS DESIGNATED 30-DAY EVALUATION COPY, THE FOLLOWING TWO PARAGRAPHS ALSO APPLY TO YOU:
THIS SOFTWARE IS MADE AVAILABLE FOR A 30 -DAY EVALUATION PERIOD ONLY. TO EXTEND YOUR USE OF THE SOFTWARE, YOU MUST CONTACT SGO OR AN SGO AUTHORIZED RESELLER TO REMIT THE PURCHASE PRICE AND RECEIVE YOUR AUTHORIZATION CODE OR NEW PAID -IN-FULL COPY ENTITLING YOU TO THE COMMERCIAL LICENSE RIGHTS GRANTED BELOW.
DURING THE EVALUATION PERIOD, YOU MAY USE ONE COPY OF THE SOFTWARE ONLY TO EVALUATE IT. USE DURING THE EVALUATION PERIOD FOR ANY OTHER PURPOSE, INCLUDING COMPETITIVE ANALYSIS, IS STRICTLY PROHIBITED. UPON YOUR PURCHASE OF A LICENSE FOR THIS SOFTWARE, USE OF THE SOFTWARE SHALL NO LONGER BE SUBJECT TO THE FOREGOING RESTRICTION.
NOTE: IF THIS COPY OF THE SOFTWARE IS DESIGNATED AS A FIXED -TERM LICENSE, A LIMITED DURATION LICENSE OR A RENTAL LICENSE, THE FOLLOWING PARAGRAPH ALSO APPLIES TO YOU:
THIS SOFTWARE IS MADE AVAILABLE FOR THE FIXED -TERM FOR WHICH YOU HAVE PAID AND WILL CEASE TO OPERATE ON THE EXPIRATION OF THAT FIXED-TERM. USE O THIS SOFTWARE AFTER THE EXPIRATION OF THE FIXED -TERM, OR ANY ATTEMPT TO DEFEAT THE DISABLING FUNCTION, WILL BE IN VIOLATION OF THIS AGREEMENT AN MAY CONSTITUTE COPYRIGHT INFRINGEMENT.

SGO


# Mistika VR folder structure

# 3 Mistika VR folder structure

This is an introduction to the folder structure that Mistika VR will create in your computer:

*Note: in some cases, old names created by older versions are indicated in parenthesis ().*

### Base folders:

During the installation (and also in the initial dialog that appears when opening Mistika VR), two base folders are defined:

- **SGO Data/Media (Material):** this is the base folder where ed media files are created.
- **SGO Data/Projects (Work):** this is the base folder where all projects are created. A project contains all the work done for a production (not including the rendered media, which is defined in the point above).

You can change the location of those folders at any time, in the **Paths** dialog that appears when opening Mistika VR. These are separate because in general they have different needs: **Project files** are small but should be on the most reliable and best-protected disk. ***NB***: make frequent backups, because this is all your work. **Media files** (rendered media and camera files) should be located in the fastest storage available. Rendered files do not require backups, as they can be easily re-rendered (as long the RENDER folder is maintained, as explained below).

For collaborative workflows, when using multiple systems or multiple Mistika products, we highly recommend that you share these folders across all systems, and use the same ones for all products.

### Project subfolders:

When a new project is created, it will contain several sub-folders. Most of them are not relevant for Mistika VR, but they need to be present for different SGO products to be able to work with the same project. The only relevant project folders and project files for Mistika VR users as follows:

- **ProjectName/DATA:** This is where your sequences are saved, including all the stitch information. All the work done on a sequence is saved in a .vrenv file. The same project may have several .vrenv files (as many as you create with the *File>Save as* menu). This permits multiple sequences of a complex production, or multiple versions of a same stitch, to exist.
- **ProjectName/DATA/RENDER:** This point is for advanced users and for programmers. Most users will not need to use these files, but they are documented here as reference.

Every time a render is done (or when *Add to render queue* is used), a render job file (.rnd file) is created in the **DATA/RENDER/RenderName** folder for each shot containing all the information that is passed to the renderer. An .rnd file is like a .vrenv file, but split into shots (one .rnd per shot). A sidecar .clp file is also created with each .rnd file. This contains the metadata of the new media file that will be created when rendering the .rnd file. In these files, you will find all the render metadata not defined in the .vrenv, including the render resolution, render path, and render format.

Apart from rendering by means of the interface, you can use the .rnd files in two further ways:

1. Mistika Command Line Interface (CLI)[130]. The **mistika -r CLI** permits the background render to be launched as a command line (*mistika -r PathToRnd*).
2. Add to render queue permits the .rnd files to be created and then user scripts identified for each one passing the .rnd and some other metadata to them. For example, integration with Smedge and other render farms[121] use those files to submit render jobs.

The .rnd files are also interesting in cases when a render must be repeated exactly as it was made (for example, if the rendered media was deleted in order to make space) for which you can use any of the above methods.

*Note: All the files mentioned above (.vrenv, .rnd, .clp etc) are just text files. You may want to manipulate them with your own scripts for better integration into your workflows. More details can be found in the chapter Mistika scripts syntax.*[132]

**Global application settings and camera presets folder (SGO AppData/VR folder)**

The software also needs a global folder to save configuration files and global settings that do not depend on a specific project. This is the **SGO AppData/VR** folder in your user account (on Windows, it is located in *%UserData%/SGO AppData*). Most of these files are not intended for direct user-manipulation, but there are some interesting files that you may need to be aware of, namely **SGO AppData/VR/etc/CameraPresets.** This contains camera presets describing all known camera rigs. SGO & Mistika VR users frequently publish new camera presets before they appear in new releases: these can be installed by simply locating them in this folder: **SGO AppData/VR/runBatch.cfg**

This is the runBatch configuration file for the *Add to render queue* button, and it is created first time that this tool is used. It is explained in the integration with Smedge and other render farms document[121].

Lastly, some configuration files are shared between all Mistika applications, and these are located in this folder: **SGO AppData/shared**

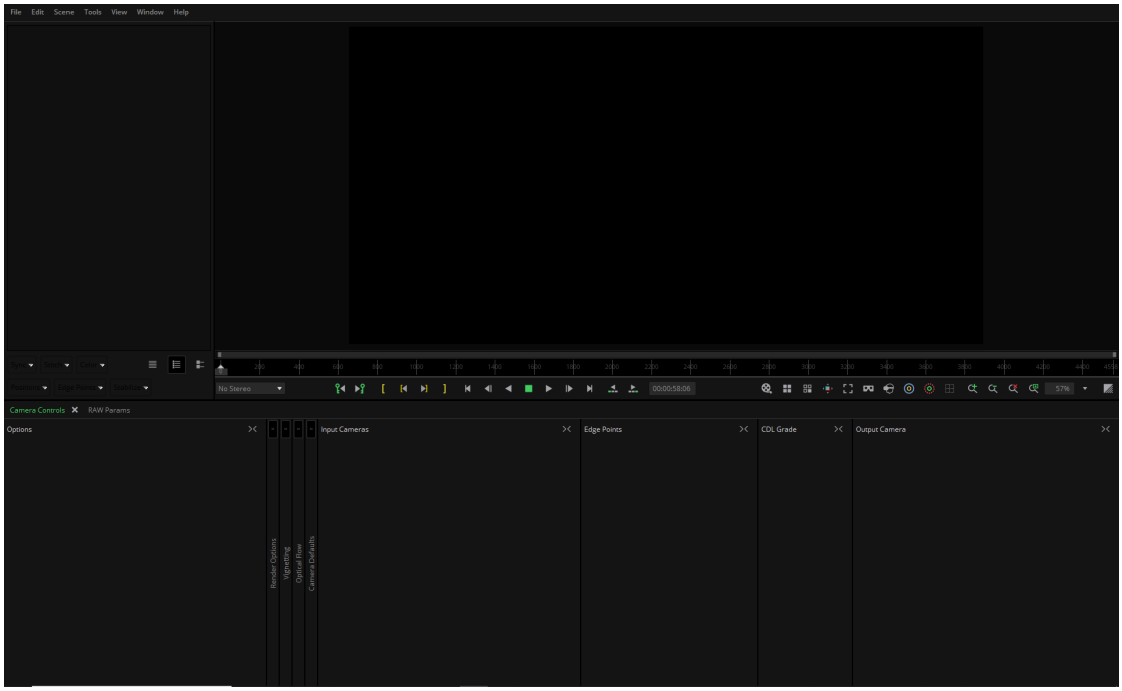
The content of this folder depends on which products are installed. In general, Mistika VR users not using other Mistika applications do not need to do anything with them except be aware that they exist, as this folder must be shared between all Mistika Technology-based applications.

SGO

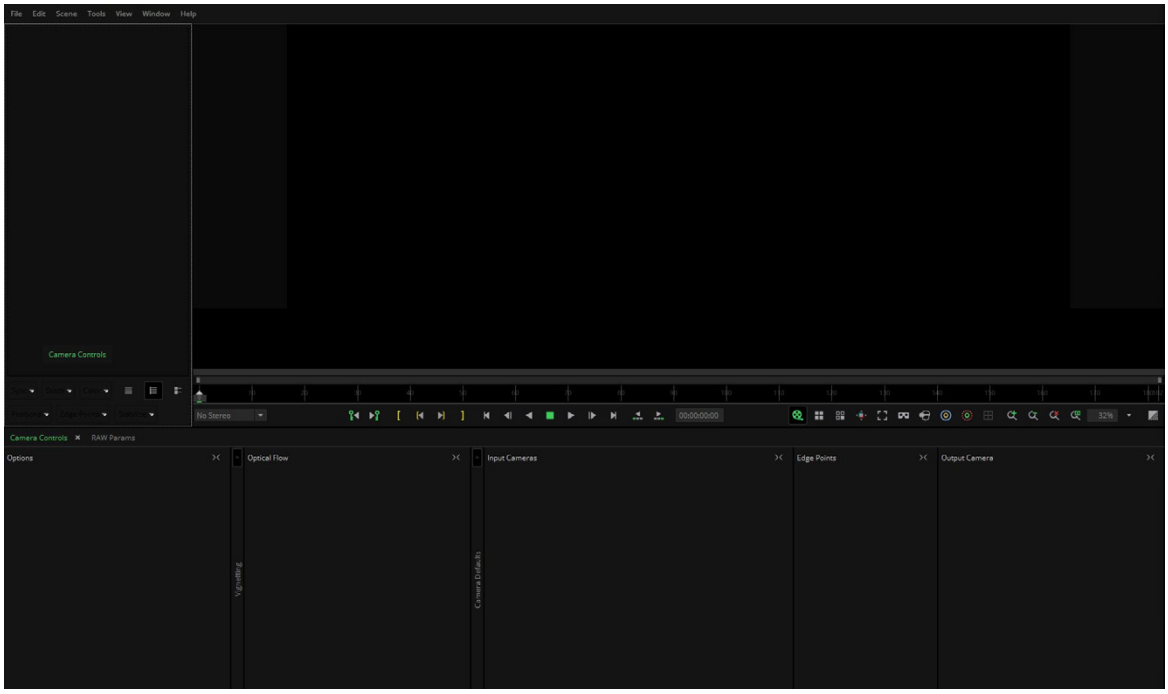

# Mistika VR Interface

# 4 Mistika VR Interface

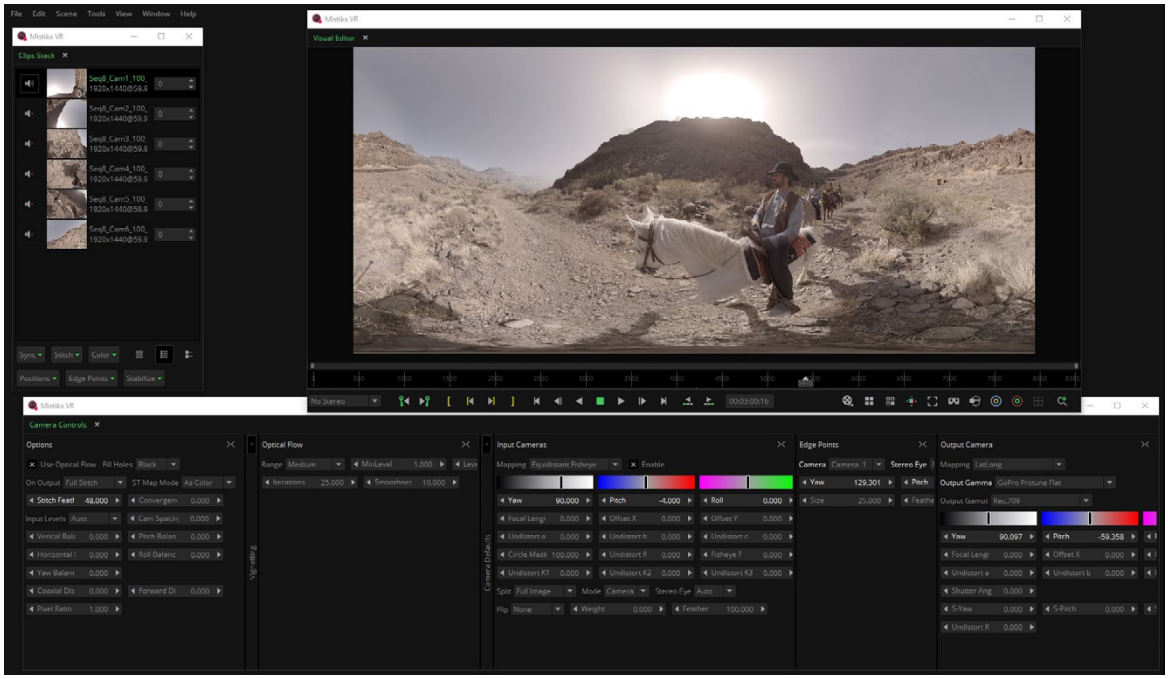
On the first launch of Mistika VR, this Interface will appear:




Mistika VR's interface is composed of dockable windows, allowing you to personalize its layout to your own taste. To do so, navigate to *Window>Autohide Tabs*. Select this option, and the panel names will change to green.



The panels in the layout can work independently:



The panels in the layout can also work together, as shown below:

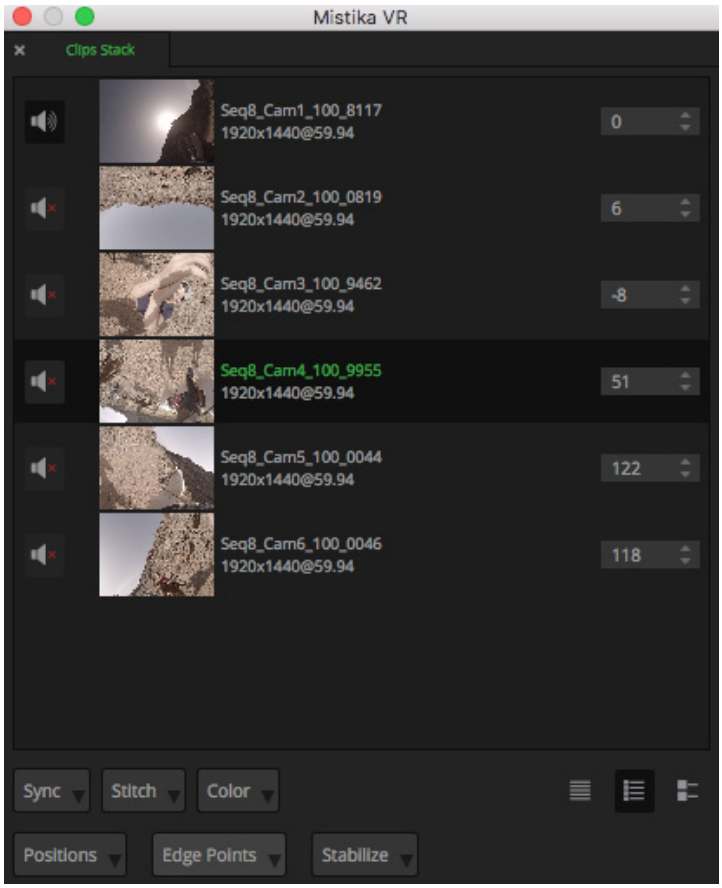
Personalized layouts can be saved in *Window>Layout>Save*. When using a saved layout, navigate to *Window>Layout>Load.*

## 4.1 Clip Stack

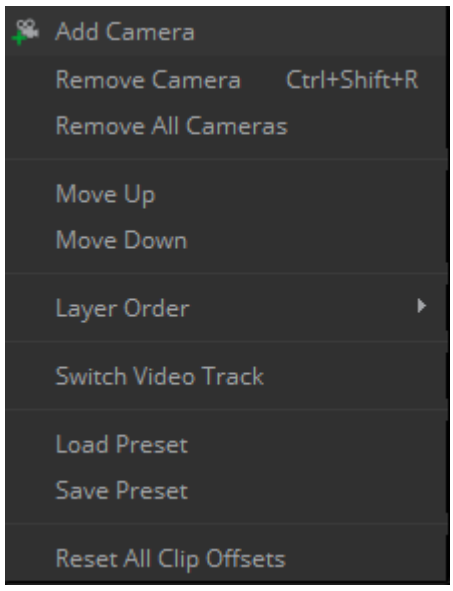
Clip Stack is a data area used for uploading clips. At the bottom of the clip stack there are also controls related to **Scene** and **Stitching tools:**

- **Sync:** once different shots are located to the same point thanks to a signal, for example a clap, this feature allows synchronization to be performed.
- **Stitch:** stitching is the act of mixing together the different shots in order to make a unique camera rig for virtual reality. See the step by step guide in chapter 6. 51
- **Color:** allows color matching for different shots.
- **Positions:** improves the places where the shots are located in the rig.
- **Edge Points:** improves the stitching within the mix of the shots. See more in chapter 10.3 Edge Points in 3D. 86
- **Stabilize:** a full explanation of this feature is found in chapter 10.5 Stabilization. 88

The cameras can easily be moved up or down by drag-and-dropping the selected clip.




By right-clicking anywhere in the Clip Stack area, a contextual menu will open, as shown below:




The options in the menu enable various different operations:

- **Add/Remove Camera** adds or removes cameras in the selected shot (marked with a white box).
- **Move up/down** moves the cameras either up or down in the clip stack.
- **Layer Order** layer system where all the cameras are placed in the level 0 by default (the stitch level). With this new feature the user can move the cameras up and down as overlays, changing the layer order of each camera.
- **Switch Video Track** allows switching to the second embedded video track of the camera.
- **Load Preset** loads the desired camera preset from the **Mistika VR Rig Preset Library**.
- **Save Preset** saves the Stitching Settings used in the shot.
- **Reset All Clip Offsets**. If cameras are not in sync, sync them either using the sync tools or manually. *Reset all clips offset* will reset the time offsets of all clips to their original position.

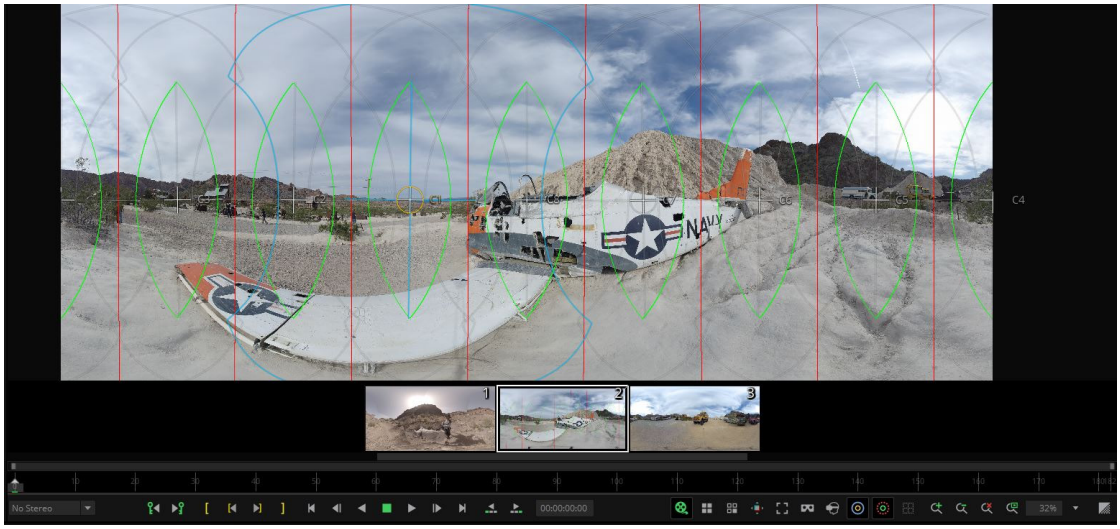
## 4.2 Visual Editor

Visual representation of the clips loaded in the clips stack, composed of three main parts: main clip representation area, storyboard, and visual editor controls, as described below.

### 4.2.1 Main clip representation area

The Main Clip representation area is for previewing loaded cameras, either in mosaic or one-input view. It also enables previewing the stitched footage:


*Courtesy of Insta360*

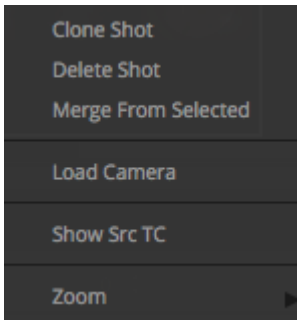


*Courtesy of Insta360*

### 4.2.2 Storyboard

The Storyboard panel permits navigation in between different shots. By right-clicking on the selected shot (marked with a white box) a contextual menu will open, as shown below:

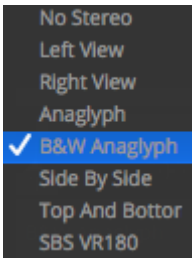
The options in the menu enable various different operations:




- **Clone/Delete** shot function serves either for duplicating or deleting the selected shot.
- **Merging several different** shots with the *Merge from Selected* tool. For more information on this tool, see chapter 14.3 Mass import of multiple shots and multi-segment shots. 120
- **Loading additional camera** by selecting the *Load Camera* option.
- **Show Src TC** displays the source timecode of the original media.
- **Zoom** for zooming-in on the Storyboard panel.

### 4.2.3 Visual editor controls

- In the drop-down menu on the left side of the Visual Editor controls section, choose between various view modes of the content shown in the main clip representation area:




- Next to the view modes, there are Key-frames navigation controls:

- Next to the Key-frames navigation controls, there are the Selection marks for selecting the section of the clip to ***Match color*** *in Time* or to be rendered:

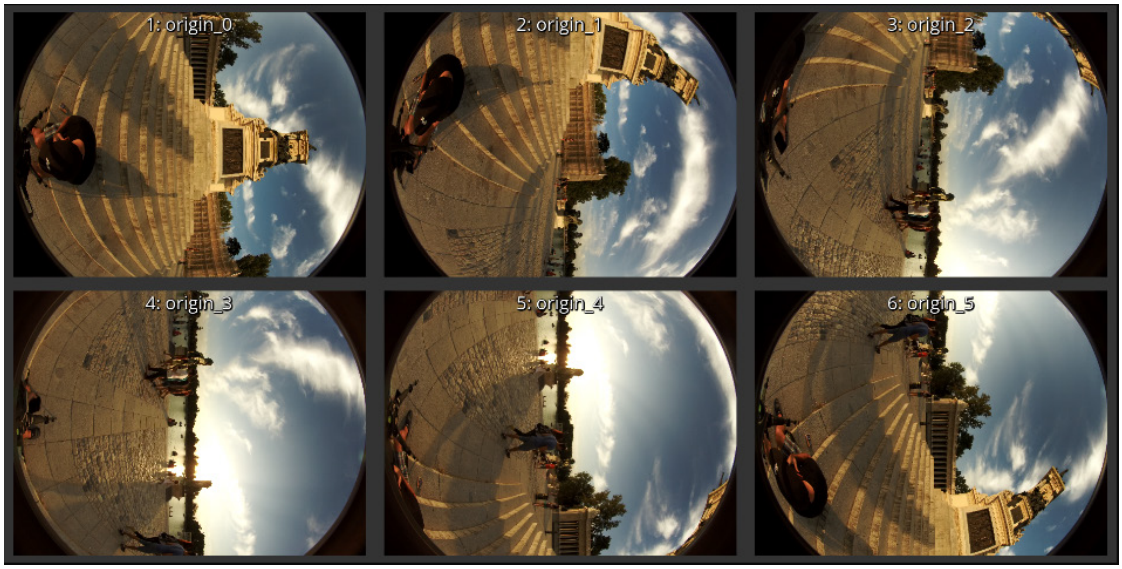
- In the center of the Visual Editor control panel, navigation tools can be found:


- In the right-hand part, there are various tools related to viewing the content in the main clip representation area:

- In the center of the Visual Editor control panel, navigation tools can be found. In the right-hand part there are various tools related to viewing the content in the main clip representation area:


1. **Storyboard** see the explanation in the previous chapter. [25]

2. **Show Mosaic** mode displays all cameras, allowing for specifying the frame where the signal needed for the audio sync is located.




3. **One Input Mode** enables the viewing of one of the cameras loaded in the clip stack, as shown below.

4. **Align Mode Tool** enables fixing vertical parallax issues. This is mainly meant for the use of VR180 parallel pair rigs. See more in Chapter 9.7. 93

5. **Full Screen Mode** allows viewing the Visual Editor panel in the *Full Screen Mode*.

6. **VR Mode** allows navigation in the 360° sphere of the stitched clip (Lat-Long view).

7. **VR Headset preview** enables a live output of your stitched VR media to VR Headsets without the need of rendering. See more in Chapter 10.8 94

8. **Camera Overlay** has two different functionalities. When the one input mode is selected, it allows lens crop. It also enables camera overlay of the stitched footage where cameras can be either switched or selected. It also enables addition/removal of Edge points.


9. **Feather Overlay Tool** is represented as the green and red line specifying the Stitch Feather.

10. **Grid Overlay** serves to specify the optical center of the selected clip.

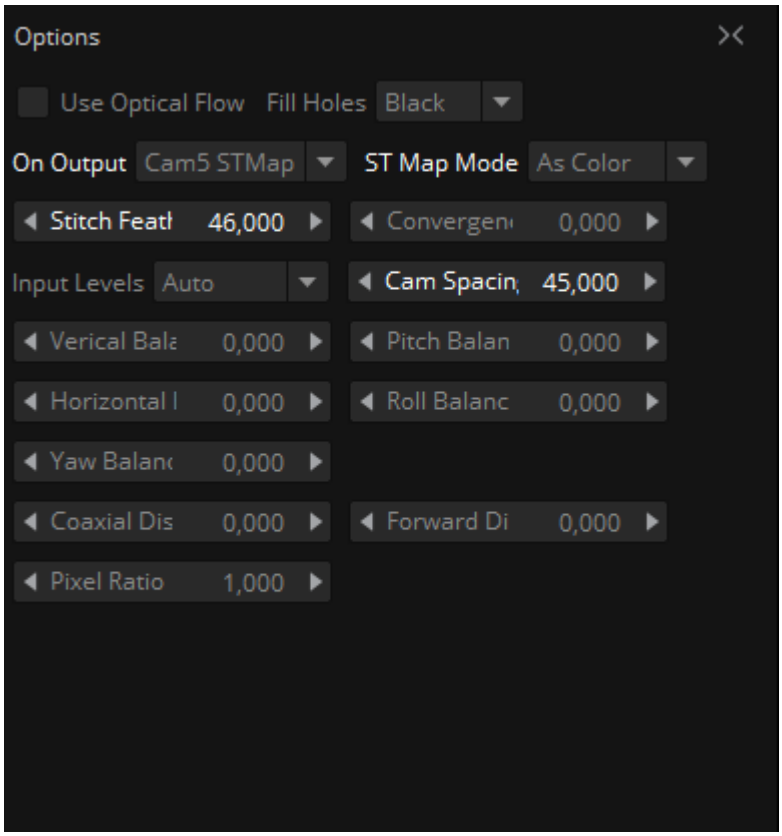
11. **Zoom in/out** reset and center Tools.

12. **Quick View** enables low-quality previewing of stitched footage. It serves as playback of footage without rendering it first.

## 4.3 Camera Controls

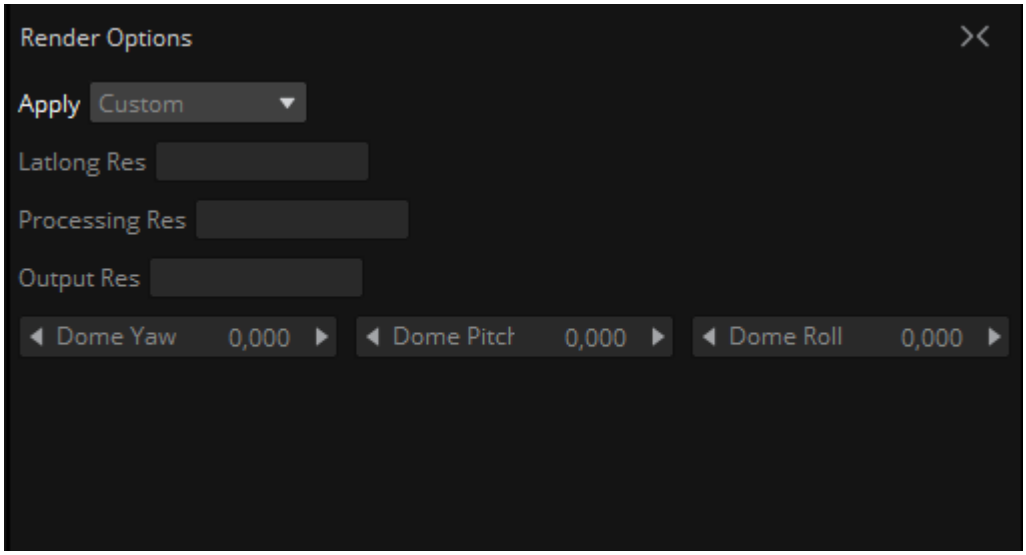
From this panel, you can control these Mistika VR tabs: *Options*, *Vignetting*, *Optical Flow*, *Camera default*, *Input Cameras*, *Edge Points*, and *Output camera*.

- **Options Tab.**




The movie files from some cameras are full data range (0-255) yet they do not state having Mistika to assume they are the default full range (16-235). This would result in excessively contrasted images, especially shadows being clipped to black. You can override this movie attribute by setting the *Input Levels* parameter into *Data (Full).*

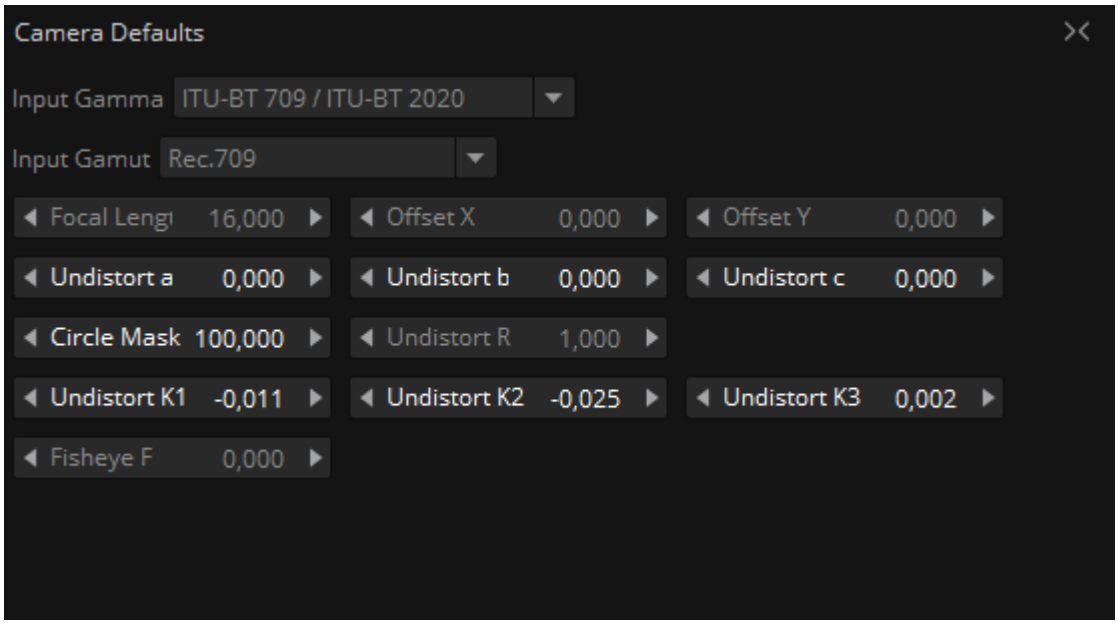
- **Render Options.**



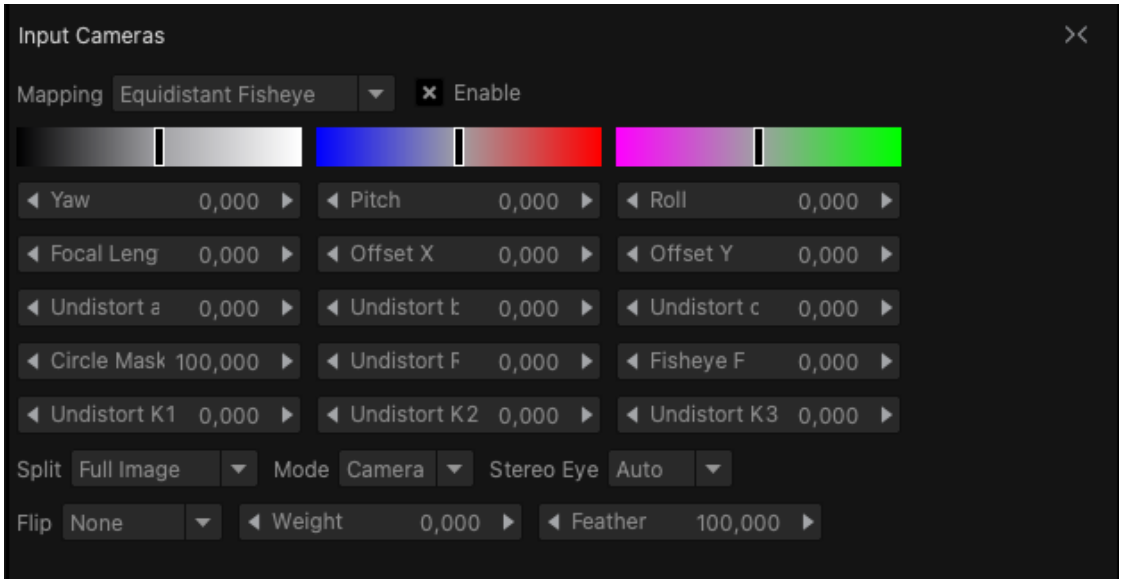

- **Vignetting Tab.**


- **Optical Flow Tab.**


- **Camera Default Tab.**

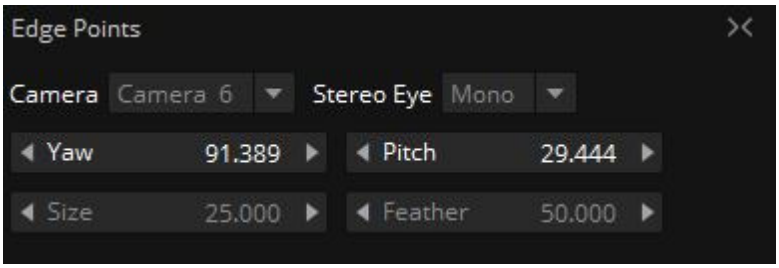



- **Input Cameras Tab.**

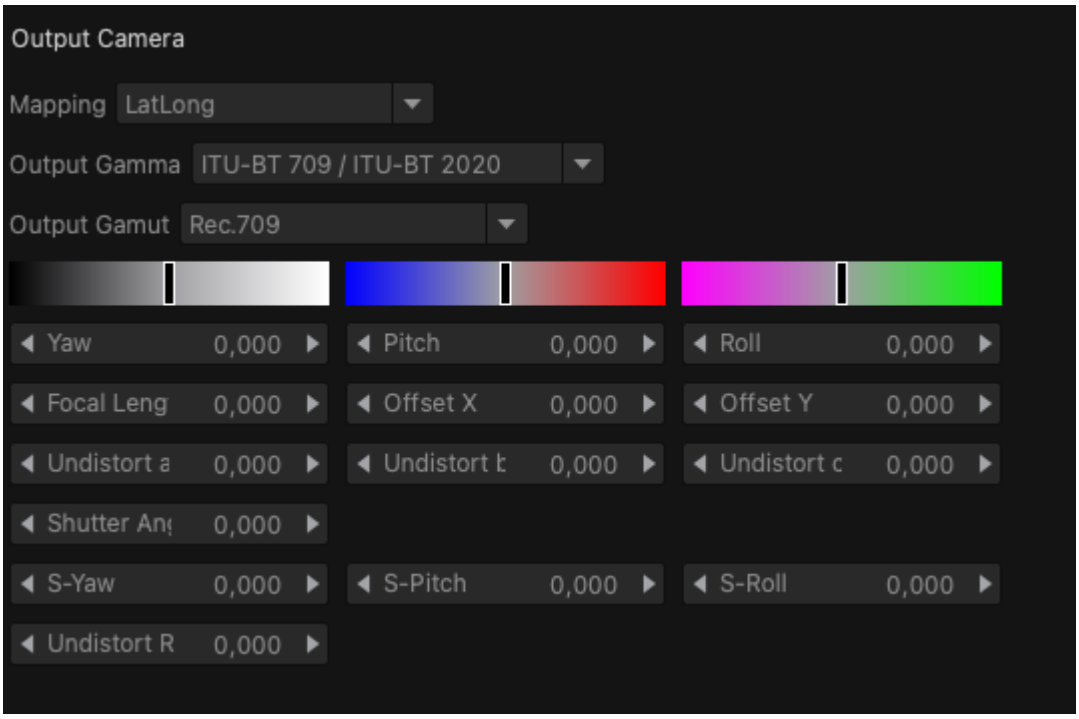



- **CDL Grade.**


- **Edge Point Tab.**

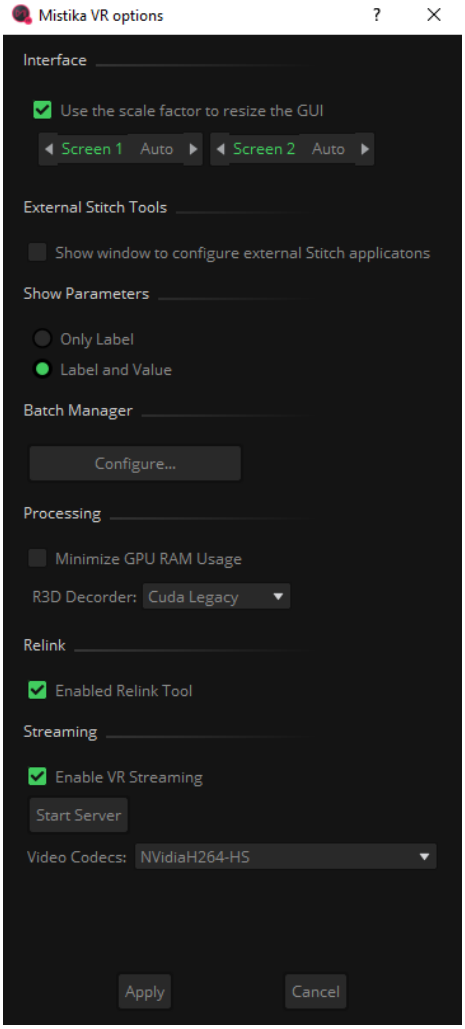



- **Output Camera Tab.**

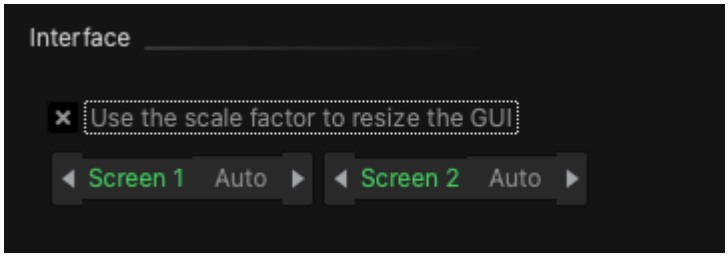



## 4.4 Preferences panel

The preference panel or Mistika VR options is located in *File* > *Options*. This menu allows to customize different parameters within Mistika VR:



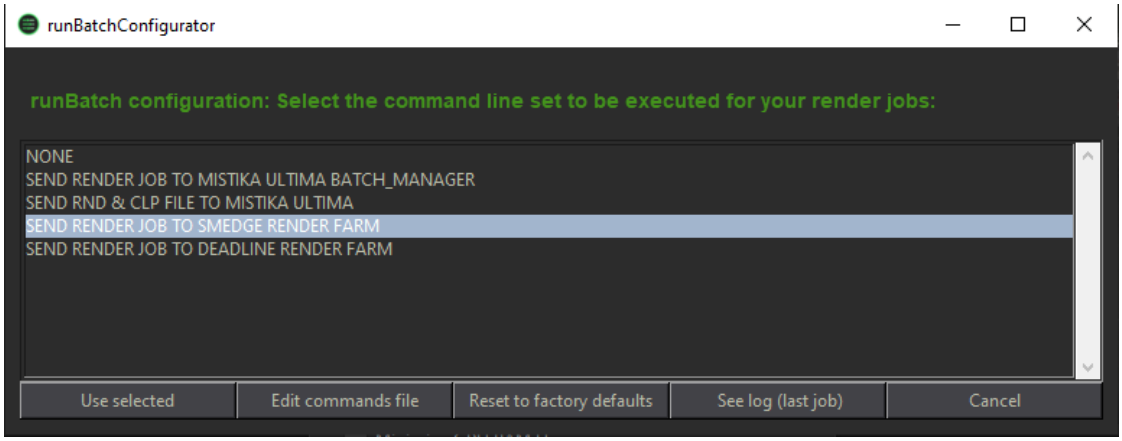

- **Interface**: this parameter controls the capability of showing correctly Mistika VR interface in high resolution monitors, improving the visibility of the whole interface. It is suggested to improve the values one by one until the GUI fits the user monitor. If there are two monitors set up, there will be one scaling factor for each one.




- **External Stitch Tools:** this parameter allows the user to decide wether to show or not to show the window that configures the external stitching

apllications such as PTGUi  or Autopano Giga. Further information is expalined in the Usage of Autopano Giga from Mistika VR[63] chapter.

- **Batch Manager:** when clicking on *Configure* a contextual menu shows up which lets the user configure which external render farm will be used for batch rendering. This render farm is activated and used in the *Add to render queue* button in the render panel permits sending a batch of Mistika render jobs (.rnd files) to external render farms, or transferring those files to user defined scripts. This feature is explained in detailed in the chapter Integration with external render farms: Add to render queue[121].

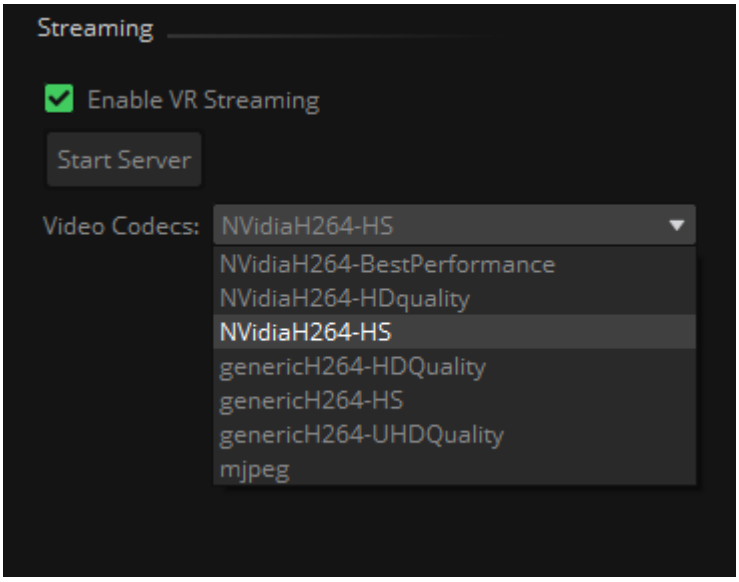



- **Processing:** it allows the user to minimize the GPU RAM Usage. It also allows you to choose the R3D Decoder that you want to use (Cuda Legacy, Cuda, OpenCL Legacy and OpenCL).

- **Relink:** it allows you the option to activate or not our relink tool when it does not find your path to your files.

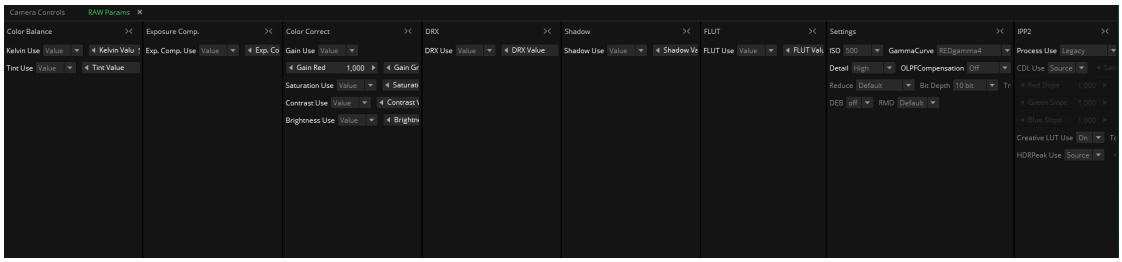
- **Streaming:** it allows you to start or stop the streaming server to establish the connection. Depending on the graphics card you can choose the video codec for streaming. If you have an Nvidia GPU, the streaming quality will be up to 4K, depending on the resolution of your project. Please note that if your project's resolution is higher, the streaming will be scaled down to 4K. If you

are using a different graphic card Mistika VR will use a generic codec that covers up to 2K resolution.




## 4.5 RAW Params

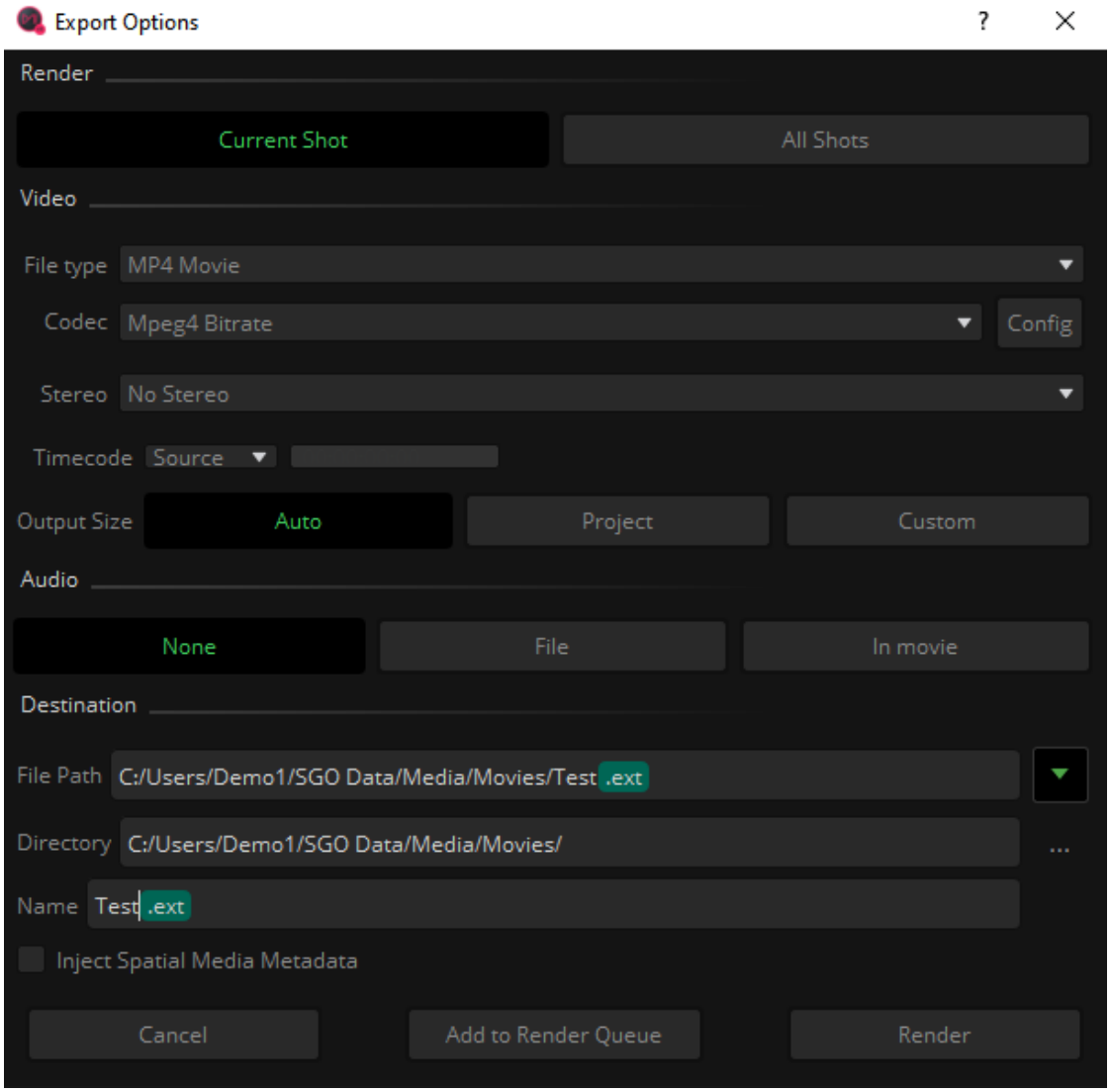
The RAW Params panel or Mistika VR options is located in *Window* > *Panels*. Media in RAW format now allows RAW data adjustments (debayer, sharpening, ISO etc, depending on camera model)



When you load a RAW file, this window will display the RAW parameters allowing you to modify them.

## 4.6 Export Options

The Export Options menu is located in *File* > *Render:*




Further information expalined in chapter 11 How to render and formats in mistika VR. 100

SGO

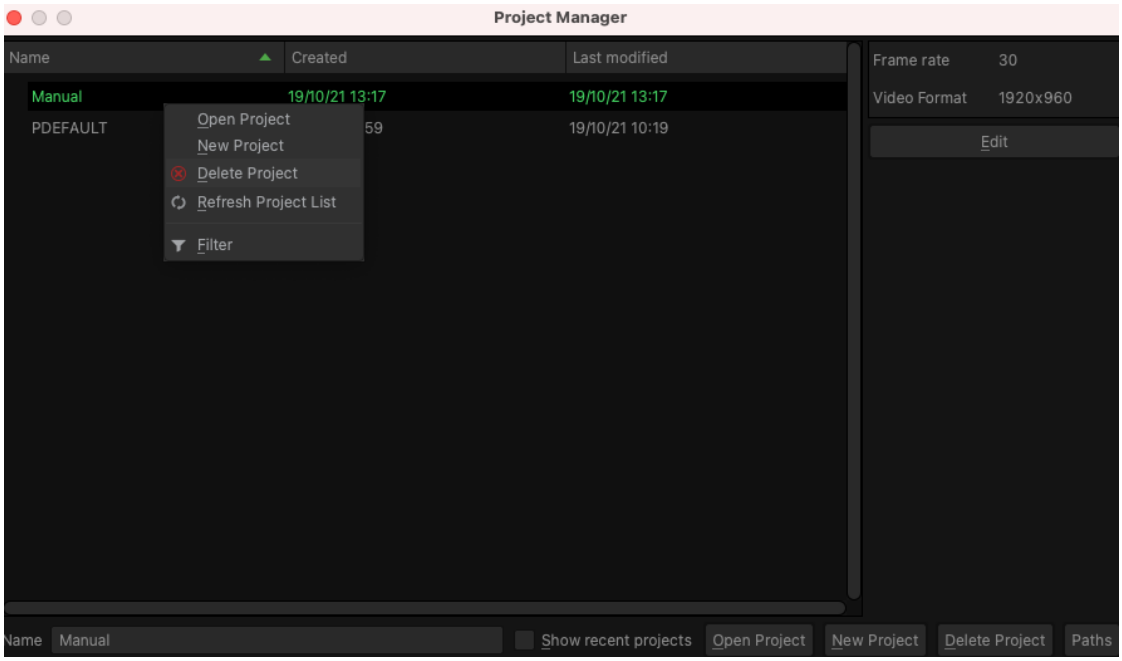

# Creating a new project

# 5 Creating a new project

## 5.1 Project Manager window

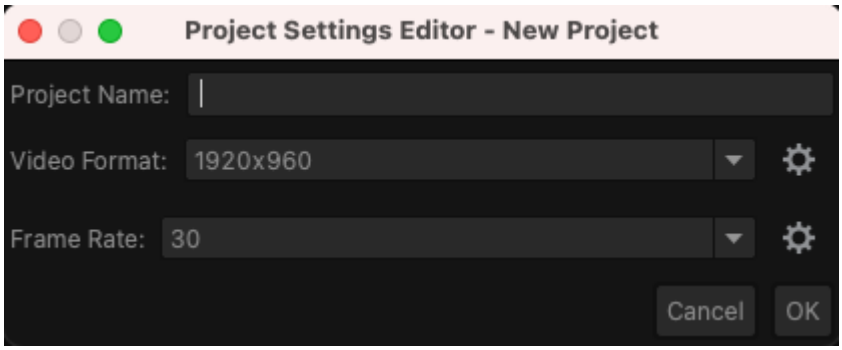
### How to create a new project

In order to create a project, follow these steps to set your own configuration. First, go to *File>Project Manager>New Project* and name your new project.



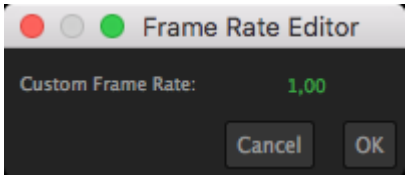

### Project/Output Resolution window

Then set the output resolution. In *Video format*, a pull down window with different output resolutions is available (scroll to navigate). Choose the one that suits you best. If you want to customize your own resolution settings, click on the wheel and a Custom Size Editor will appear where you can specify your own height and width. Click *OK* to confirm.




### Frame Rate Editor window

Next, set the frame rate of your project. Open the pull down window and scroll down to select the different options, or customize your own frame rate.




The resolution and frame rate of the master format to be produced (and rendered) are both set in the first panel that appears when opening Mistika VR. There are two types of resolution fields and frame rates that you will see in Mistika VR, which should not be confused:

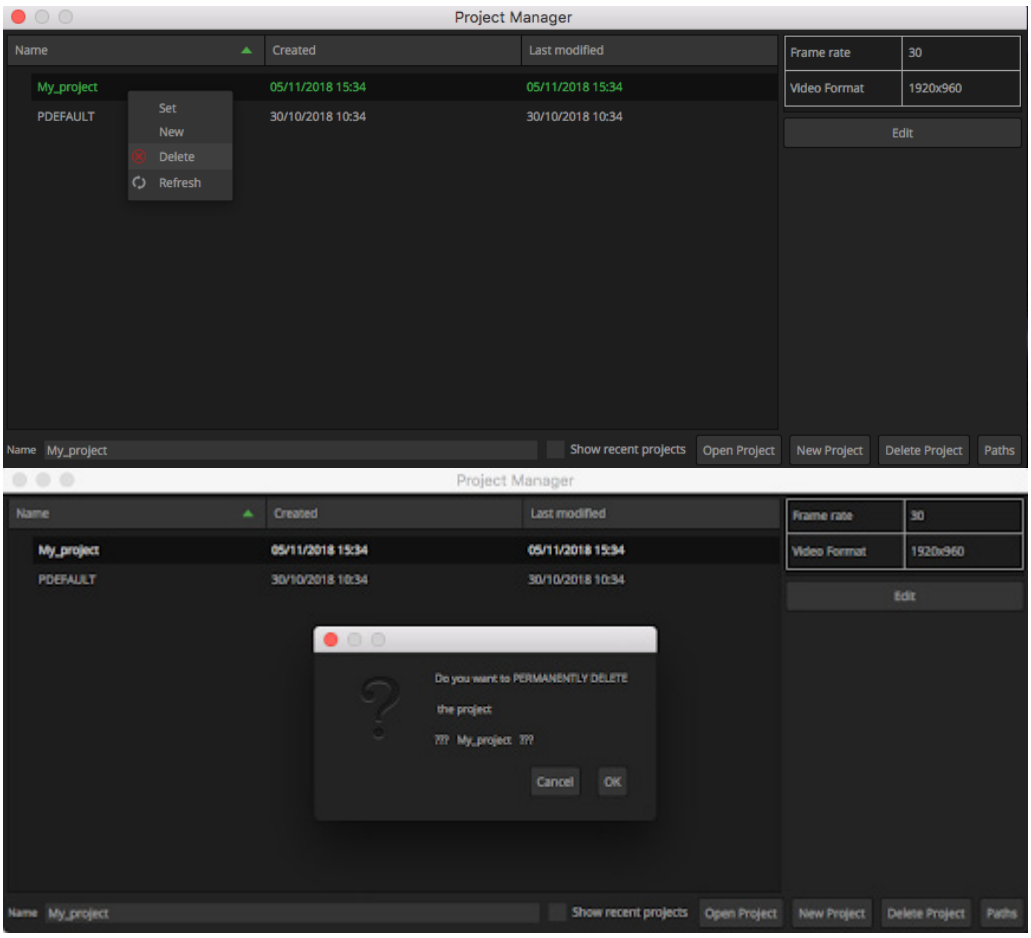
- The resolution and frame rate of each camera, which is not decided in Mistika VR (it is determined by the source clips). These can be any resolution and aspect ratio. Be aware that in many camera rigs, the cameras are employed rotated through 90 degrees to provide more vertical range, so the true camera resolution is the reverse of what is stated.
- The Project Resolution and the Project Frame Rate: these are the frame rate and total resolution that you will render out after stitching all the cameras. These ***are*** decided by the Mistika VR user, and they can be changed for each Mistika VR session.

This resolution is always forced to a 2:1 aspect ratio. This is the standard for VR 360, because equirectangular representation must be 2:1 in order to avoid incorrect distortions when mapped to the sphere, and also to produce valid geometry for Stereo 3D. The Project Resolution and Frame Rate are shown in the first panel that appears when opening Mistika VR, and once selected, it cannot be changed until the next Mistika VR session.

Mistika VR offers resolution presets for all known VR Headsets, but if for whatever reason a different resolution is required, you can also add a new format. You will be asked the new width, and the height will be calculated automatically (to maintain the 2:1 aspect ratio as per the VR 360 standard).

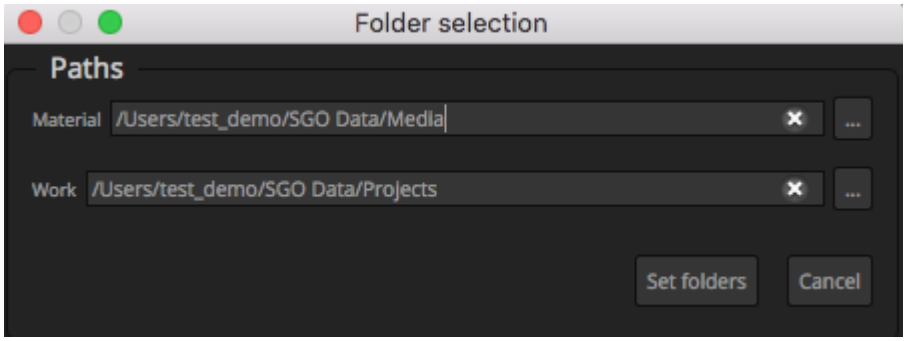
### Deleting a Project

You can delete a project by right clicking on it. A contextual menu will appear with a *Delete* option. After clicking it, a prompt will ask you if you want to delete it.



### Project Paths window

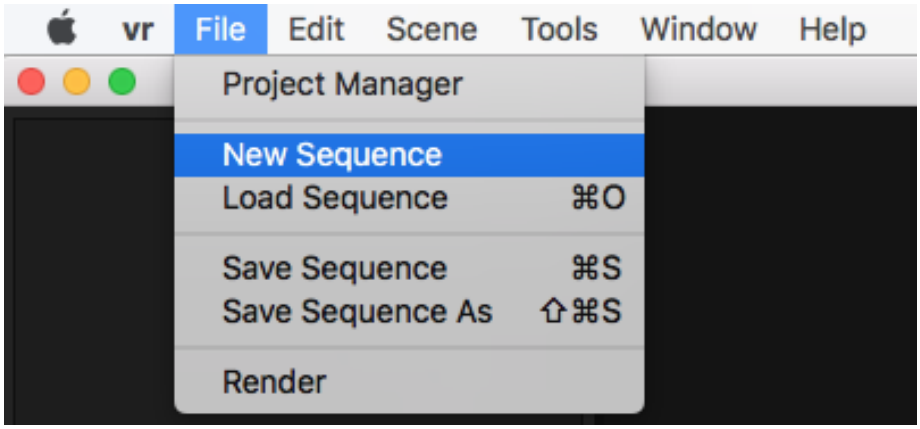
You can set the project path in *File>Project Manager>Path*. If a path is changed, make sure you have permissions for writing on the folder selected. If the selected path is not accessible, Mistika VR will not be able to create the project.




## 5.2 Creating a new sequence

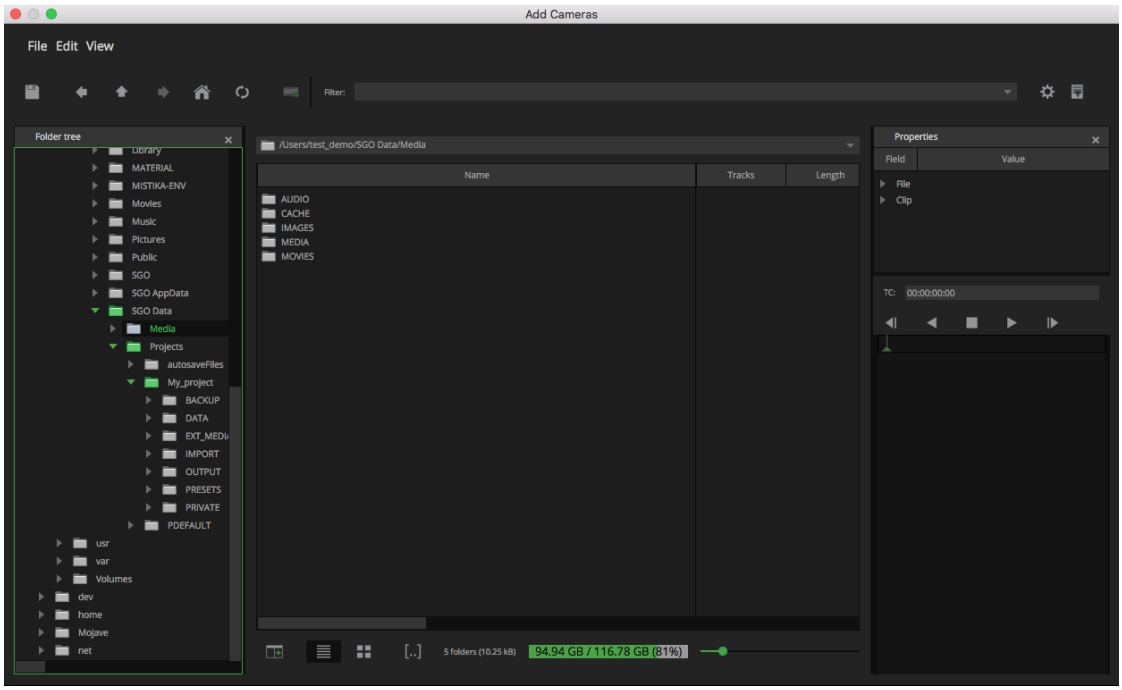
### How to create a new sequence

Create a new sequence in *File>New Sequence*.



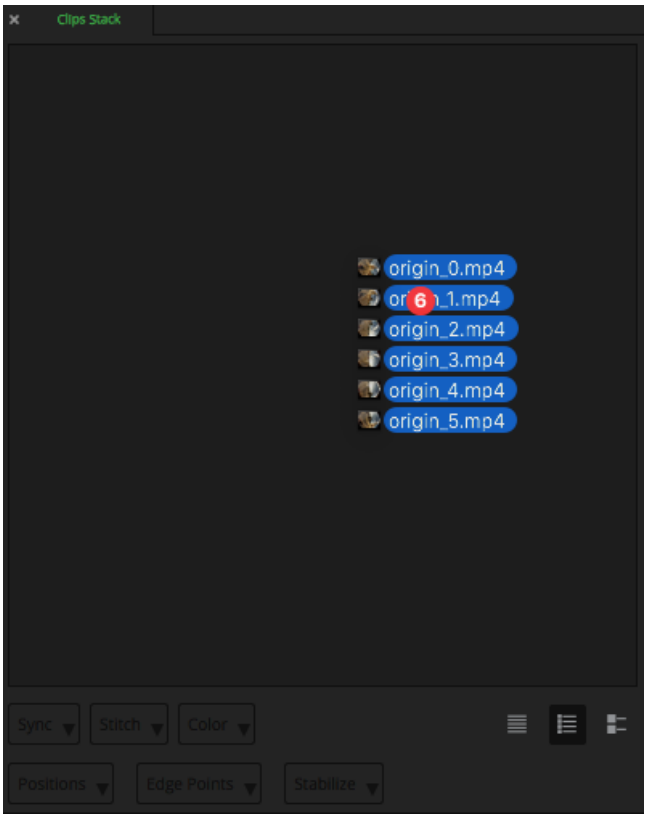

### Import Media 'add camera option'

To import media, go to *Scene>Add Camera*, or right-click, and a contextual menu will appear. Click *Add camera*. Search for your footage. Select the footage you want to import, right-click and select *Load*.




### By drag & drop from the Operative System Browser

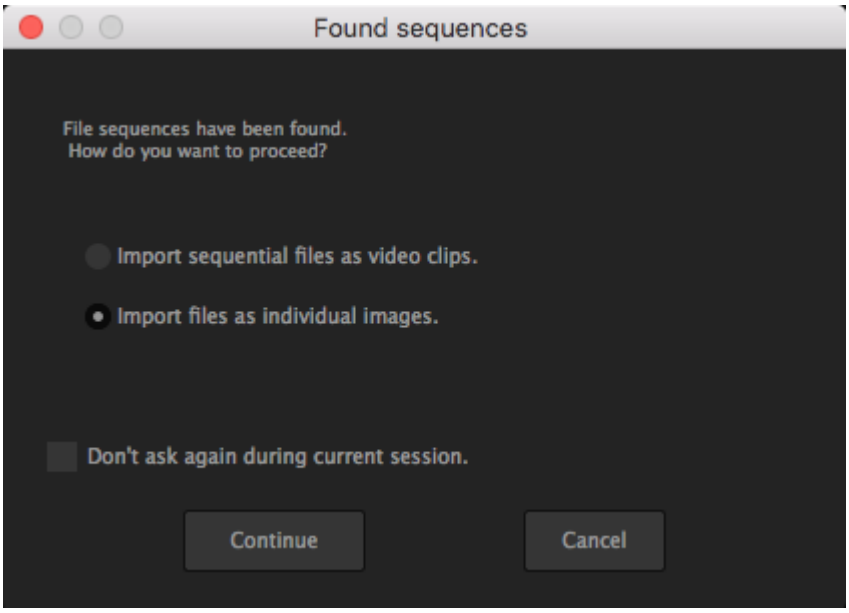
You can also import media by drag-and-dropping the selected footage from an operative system file browser.



### Found sequences window

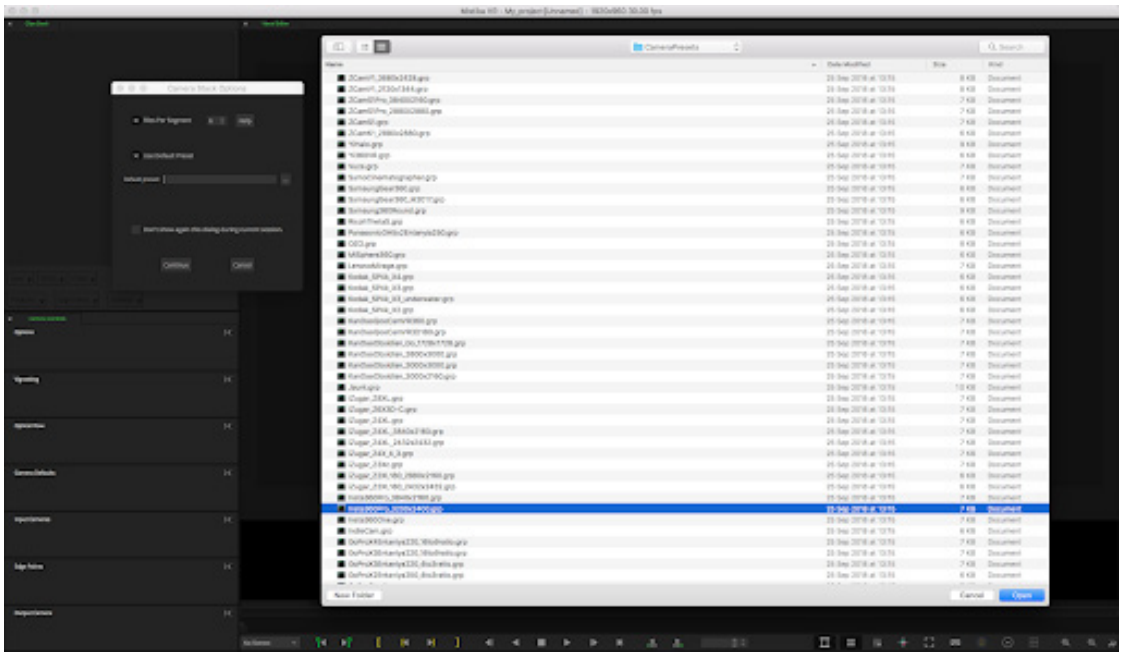
When you import media, the Found sequences window will offer you two different ways of importing:

1. Import sequential file as video clips: with this option, only one clip is imported into the Clip Stack.
2. Import files as individual images: with this option, as many clips as the sequence has frames are imported.



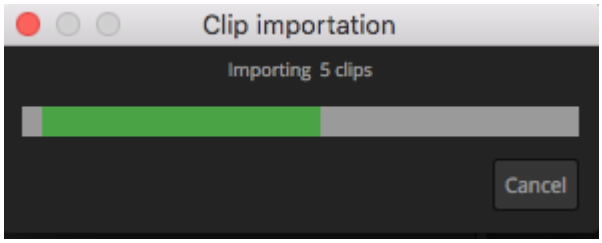

### Loading a pre-built preset in the import media process

There is also the option of importing the camera rig preset in the import media process. Click on the three dots to launch the Preset Library. Select the preset that suits your media. Click *Continue*. By following this media process steps, the stitching process is also carried out.



**Progress bar clip importation**

Wait until the Progress Bar clip ends.



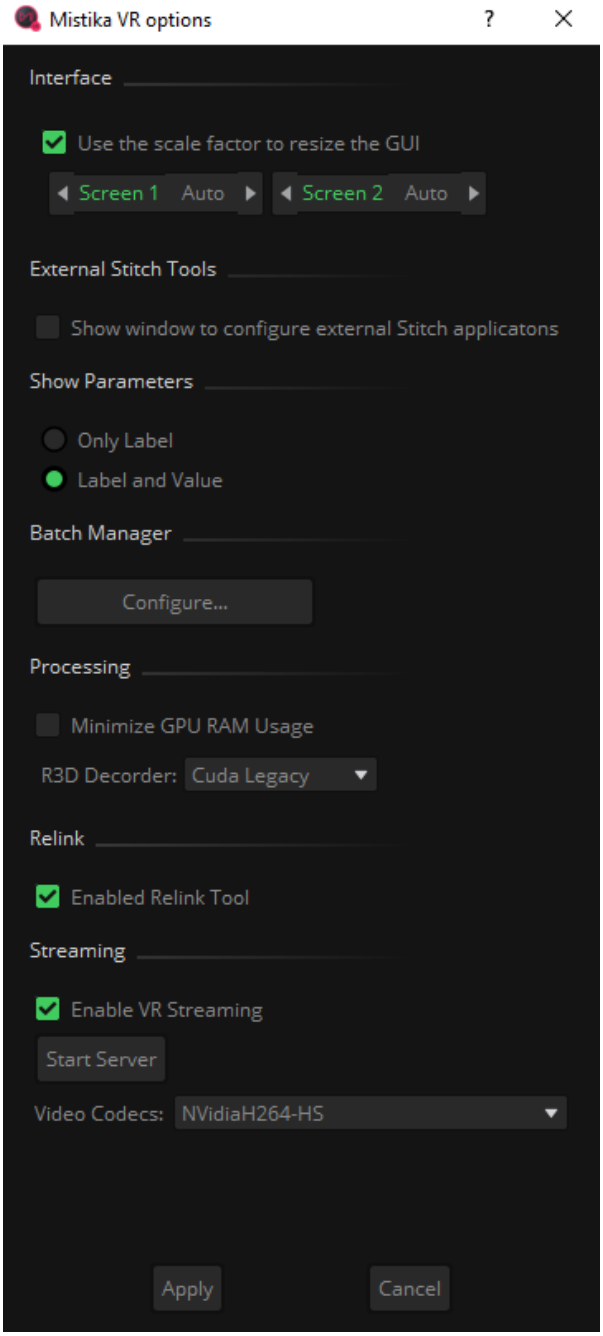

## 5.3 Relink tool and path translator

### Activation of the Relink Tool

When a media file is not found (for example, if it had been located in a different path in the original system) then the Mistika application will ask the user for the new location. Once the user has answered this for the first clip, Mistika will save this information in a configuration file and try to use it as a template for cases arising later.

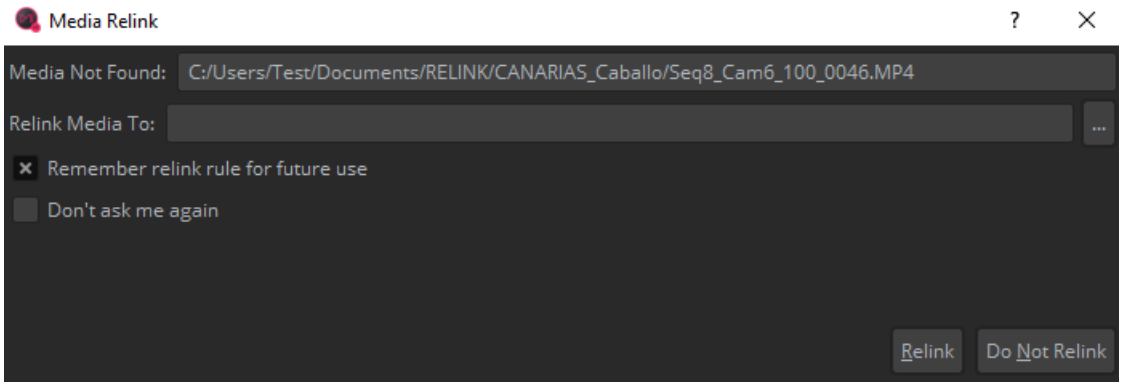
In order to activate the relink tool, there is a check-box inside the Mistika VR Options menu, as seen in the screenshot below. When this options is applied, every

time MIstika finds missing media, it will ask the user to relink them. This option allows the user to decide if he or she wants to leave some media unlinked in the project, but there is no desire for the Relink Tool to appear every time Mistika VR is opened.

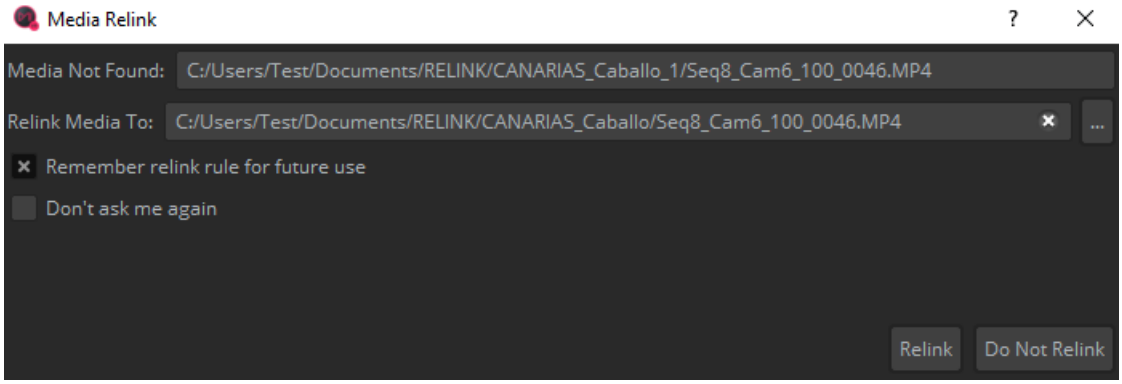



**Relink tool**

As long and the Relink Tool is enabled, whenever the user moves his or her projects files from one location to another, Mistika VR will ask to relink the media because the footage was referring to the previous folder. The media offline will appear as a checkboard and the Media Relink window will pop up:




The relink camera will ask you for one clip that is offline and what new path to refer to. Now, the three dots button of the "Relink Media to" must be clicked which will open a Media Browser. Afterward, the exact shot has to be selected, and then Mistika will analyze the rest of them and automatically relink all the footage.




It is advised to keep your shots in the same folder in order to facilitate this process. The two checkboxes available allow the user to decide to remember relink rule for future use, which means that if some media is not found, it will ask for that media after one is selected and linked. Or if the user does not want to pursue with the relinking by clicking on "Don't ask me again".

**Path Translator**

The relink tool works as a path translator in order to move projects or media between different operative systems, such as moving from Window to a Macintosh. The easiest way to configure a multi-platform environment using different paths on each system is to transfer an example project to the others and answer the prompts when loading it. Mistika will then learn from these examples for solving future cases and no more prompts/questions should appear.

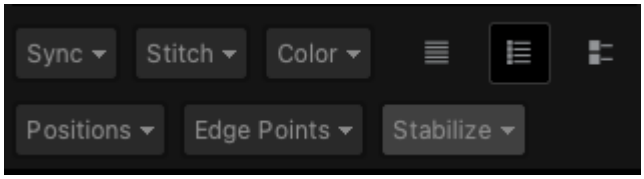
SGO

# Stitching in Mistika VR: a step by step guide

# 6 Stitching in Mistika VR: a step by step guide

This chapter contains a step-by-step guide on stitching in Mistika VR using footage shot with a GoPro custom made VR rig. It provides a general view of the stitching process in Mistika VR and the most important features to be used. Notwithstanding, each tool will be explained with details in the following chapters of this manual.

As seen in the screenshot below, the VR tools are organized on a friendly user way, ordered from the first one that should be used, the Sync, to the last one, which is the Stabilization:



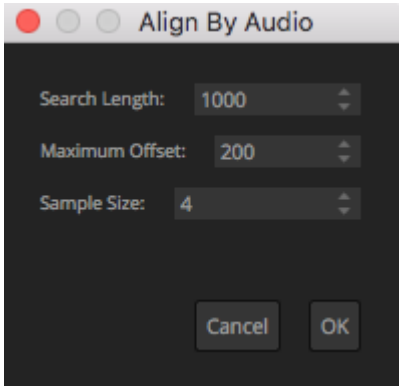

### 1. Import the Media

First of all, import the media, as it is explained in the previous chapter 43.

### 2. Sync

Once the footage is imported, look for the audio signal in order to carry out synchronization. This step is particularly important in GoPro rigs for stitching to be successful. To do it properly, navigate to the frame where the clap (or other sound signal) is found. *Mosaic* mode helps in this process of locating the correct frame.

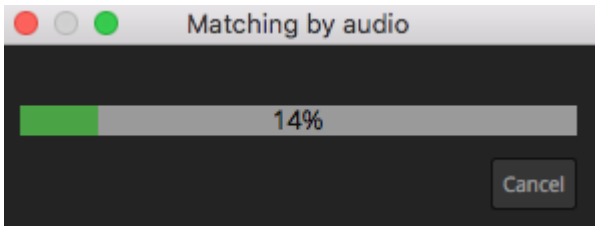
Now, in the *Sync tool*, click on *Audio Sync*, which will pop up the *Align By Audio* window:



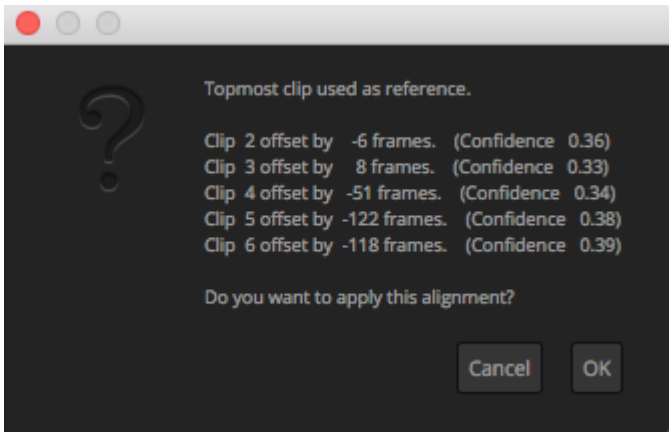

The parameters contained in this tool work as this:

- *Search length* is the length of the sound sample to be compared, centered at the current frame position.
- *Maximum offset* is how much the cameras may be off sync, to avoid false matches at unlikely large offsets. If you turn on the cameras one by one, think how many seconds it may take, and specify that number as the maximum. Set the current frame at a zone where there is some identifiable noise (eg. a clap, people talking), and run the match.
- *Sample size* should be left on default. The audio needs to be split into parts - 'windows' that can be matched - and this is the size of these parts: smaller means more precision on transients, but lower frequencies may get ignored with very small windows. Maintain a range of 2 - 4, which works best. This field should probably be hidden, as it is too technical, and tweaking usually does not produce significant changes.

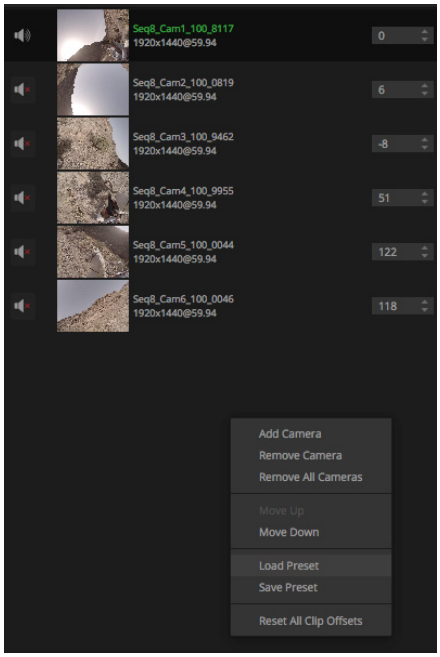
Once clicked on *OK*, the process starts, a progress bar will launch.



When the process finishes, a prompt will ask you to apply the alignment. Once applied, the offset will be activated in the Clip Stacks menu, where the user may decide to move manually the frames in case of need for specific adjustments of each camera.




### 3. Applying the preset




Now apply a preset from the Mistika VR preset library. Load it by right-clicking on the clip stack which opens the library. Select the Preset for your rig. More information about preset is explained in detailed in the following chapter[61].

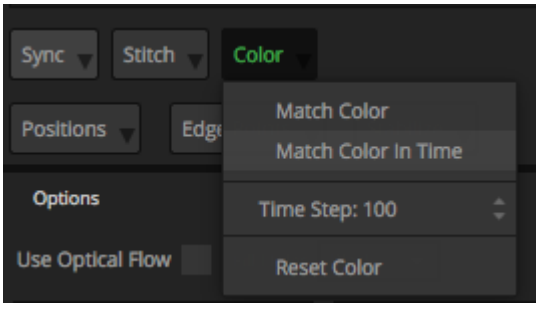
### 4. Adjusting the horizon

After applying the preset, the result is an image like the one below. Now align the horizon with *Alt + drag*.

The best practice is to find a vertical in the image and fit it with the yellow arrow.

### 5. Matching colors

Match color will automatically adjust the color of all the cameras to look the same. *Match Color* will only match for the current frame selected. *Match Color in Time* will start the process of matching the entire shot, or the range limited by the In and Out points in the timeline editor. Time Step can be used to match color a decided range of frames. Finally, the tool allows to reset all color modifications.

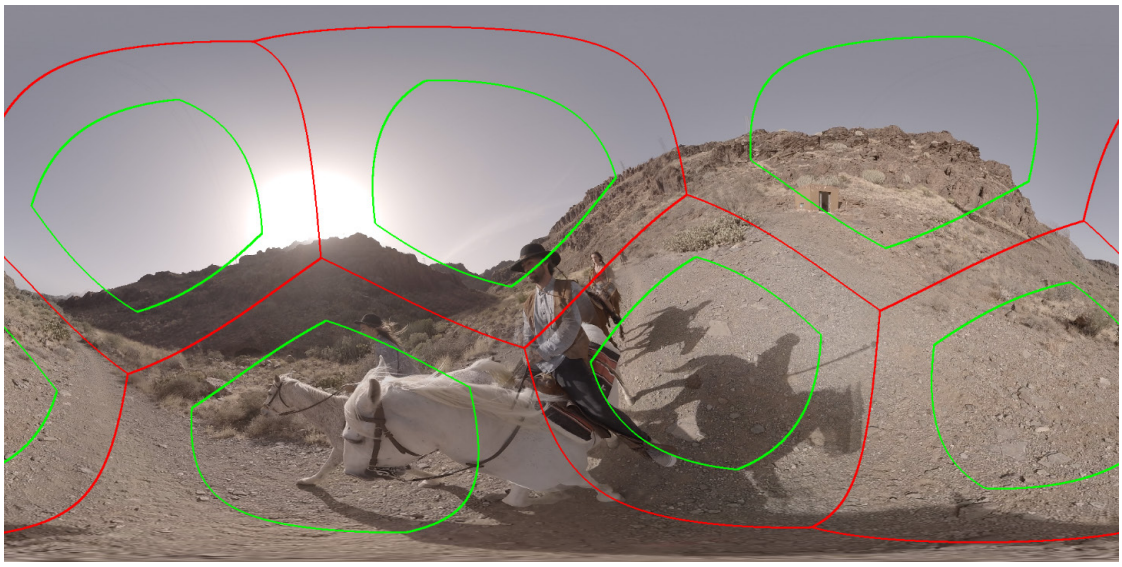


### 6. Feather Overlay

The Feather overlay is a visualizing mode that indicates with red lines the position and crop factors of the cameras. It is activated in the *Storyborad* with the shortcut 2 or by clicking on the following icon button:

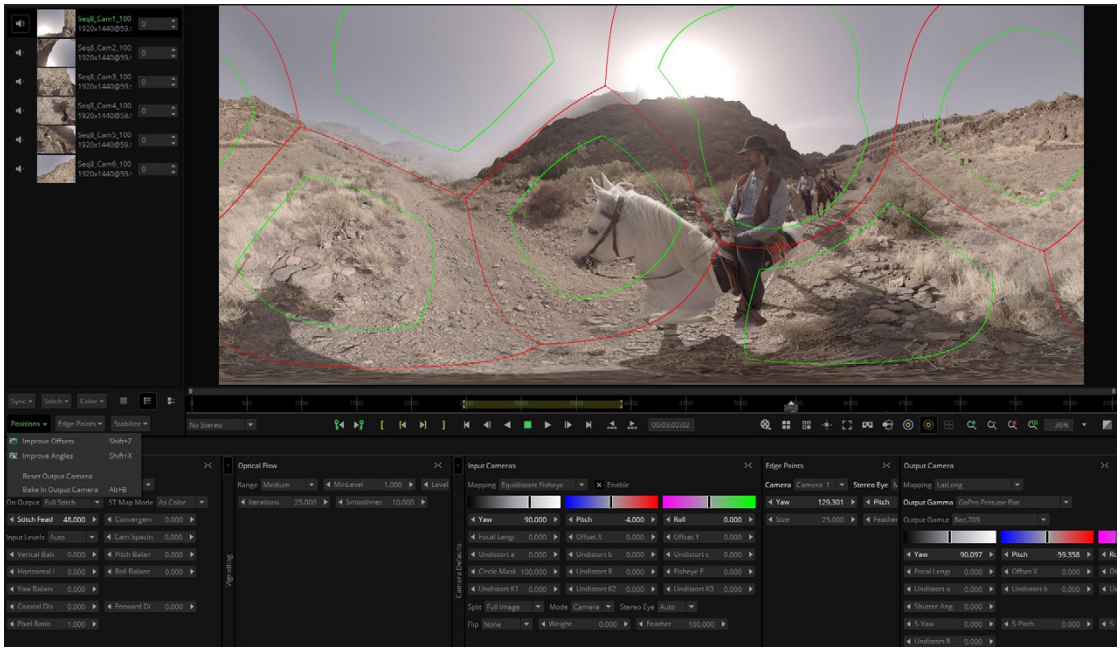
### 7. Stitch Feather

The *Stitch Feather* located in *Camera Controls* is the parameter the adjust the overlay between different shots. It is indicated with green lines. It creates a smooth area between the different cameras that eliminates artifacts in collaboration with the *Optical Flow* parameter.



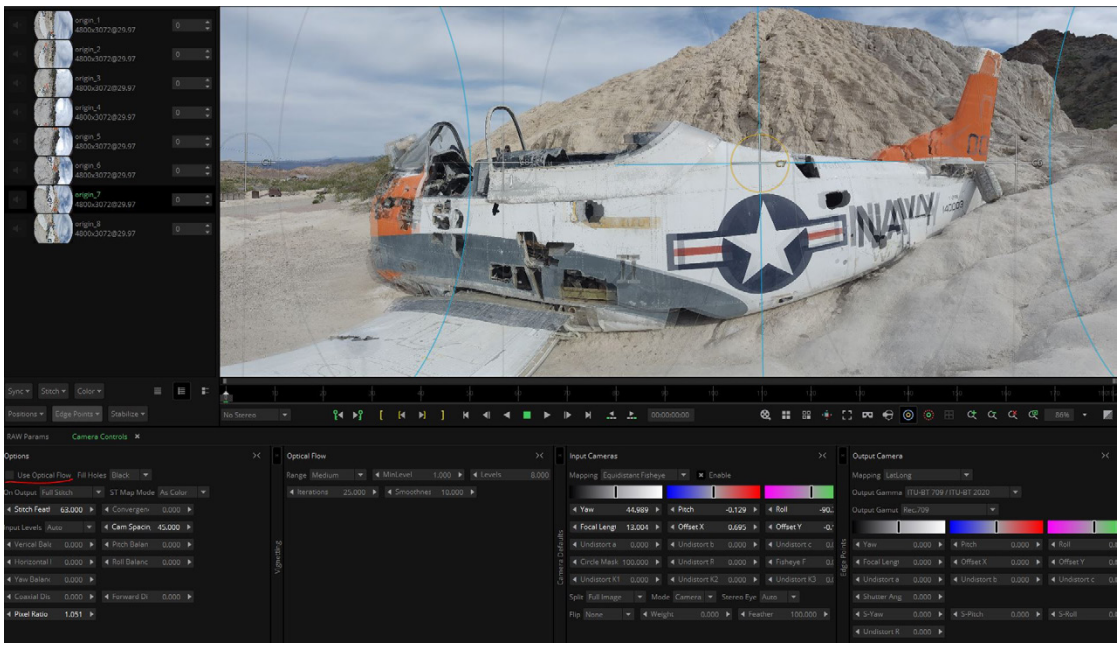
### 8. Improve Positions

The *Positions* tool contains two features: *Improve Offsets* (Control + Z) and *Improve Angles* (Control + X). They improve the places where the shots are located in the rig. It is advised to use each tool a couple times, until there is not apparent improvement in the image.



## 9. Optical Flow

The *Optical Flow's* parameters identifies every small feature in two overlapping images of a scene, establishing the exact pixel correspondence between them. Having this information allows Mistika VR to provide very precise stitching by rebuilding the edges of both images to make those features coincide. *Optical Flow* is explained in detailed on chapter 9 (79).



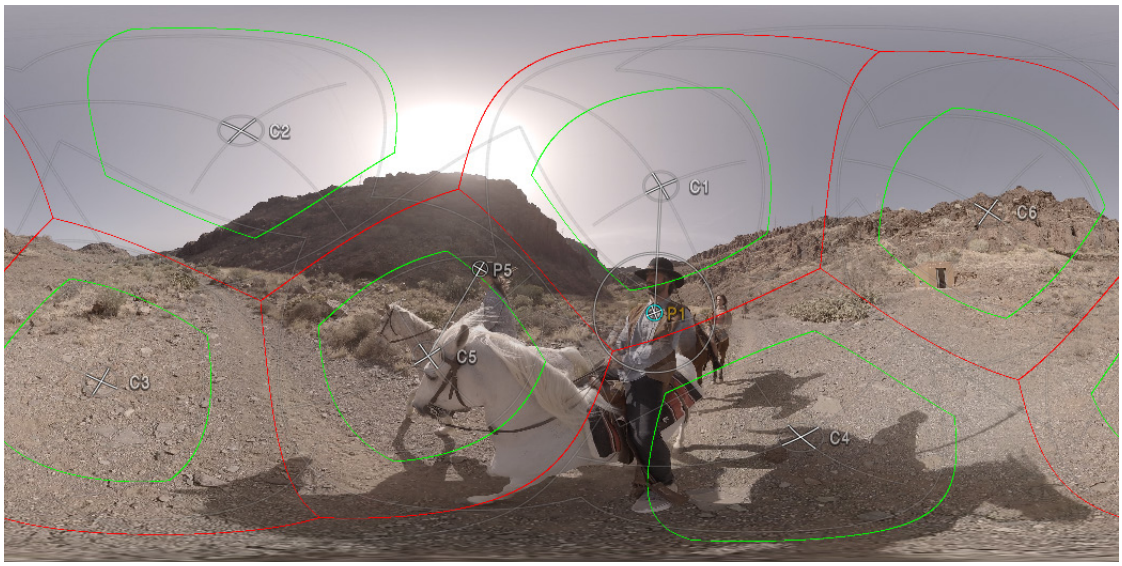
*Optical Flow Off (Courtesy of Insta360)*


*Optical Flow On (Courtesy of Insta360)*

## 10. Edge Points

The *Edge Points* help avoiding artifacts of the *Optical Flow* and 360 riggings, isolating parts of the shot that can be keyframes. They basically widen the patch around any camera, in order to push the problematic stitch lines to areas where they matter less, such as from the middle of a face to a wall next to said face. To add one *Edge Point,* activate the camera overlay, then select one camera and click on *Add Edge Point* or use the shortcut Control + Shft + A. Then, a controller will appear in the camera overlay editor, which can be positioned and widen by holding shift and moving the mouse from left to right. *Edge Points* can be key framed, in case of working with the Professional Edition of VR. The keyframing tool is explained in the chapter Keyframing. 92

Further information of *Edge Points* is explained in the chapter Edge Points in 3D. 86




## 11. Stabilization

The stabilization feature removes the inconvenient jitter and unpleasant movements of the camera. Only available on the professional edition of Mistika VR. Fully explained on the chapter 10.5 Stabilization. 88

SGO

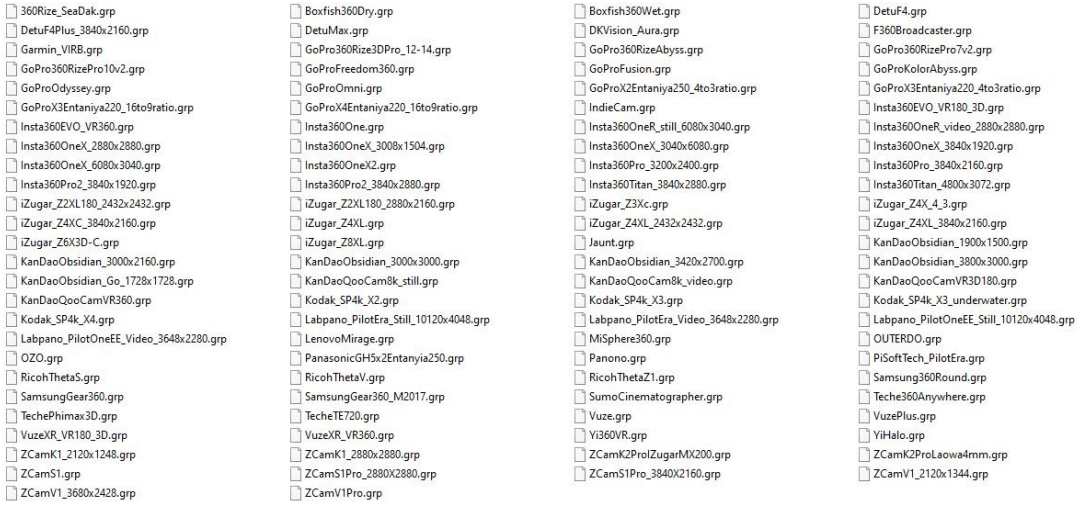

# Camera Presets

# 7 Camera Presets

Using extensive, intelligent presets, Mistika VR obtains information referencing camera position and sequences and stitches the images together.

If the rig used does not appear in the SGO library of presets, one can be created by following the instructions in Chapter 7.9 What if my camera rig model does not appear in the camera presets list? 73 If you have any difficulties creating it, send SGO a sample frame and we'll build the preset for you: go to Support Portal and submit a ticket.

Mistika VR is provided with more than 60 pre-built presets:




## 7.1 Scene suitable for calibration

First, obtain a scene which is suitable for calibration, such as one of the following:

- A scene with nothing/nobody closer than 3 metres to the camera.
- A street with buildings and trees, squares, car parking lots, or open space in a park.
- Large halls, such as sports venues, with a structured ceiling, with camera in the open.
- A public garage with no cars or columns closer than 3 meters to the camera.

If a calibration shot cannot be obtained, choose the shot most fitting the above indications. Once a suitable scene is obtained, stitching or calibration can be done in

three different ways: using Mistika's own **Improve** tools, the PTGUi, Autopano Giga or one-click calibration for Kandao, or Insta360 Pro V1 and V2 footage.

## 7.2 Calibration Scene for Rigs with Little Overlap

When using cameras with an overlap, look for a static scene with two shots. The second shot must have the rig rotated by half of its camera spacing: if there are 4 cameras at the horizon at 90 degrees, rotate the rig 45 degrees. This provides an apparent doubling of a number of cameras and produces better detection of lens distortion.

Be aware that if a camera allows different aspect ratios, such as GoPro, which can record both 4:3 and 16:9, settings for both resolutions should be created. Incorrect ratios can make stitching impossible: if set to 16:9, the GoPro Omni rig will leave holes in the stitch, so be sure to set it correctly.

## 7.3 External Calibration

- Import the clips from one shot. If the rotated shot is available, import it into the same stack (so if the rig had 3 cameras, now we have 6 camera layers).
- Scrub to a frame where the photographer walked away until hidden or at a sufficient distance.
- Export an image set to PTGui or Autopano Giga.

## 7.4 Determining the Rig Geometry in PTGui

### In PTGui Version 10 and 11

Export the frames set to PTGui. In PTGui, follow these steps:

- In the *Camera/lens data* (EXIF) popup window:
  - set *Focal Length* to 1 (this will be changed later)
  - set *Crop factor* to 1
  - select *OK*.
- In *Camera/lens parameters*:
  - uncheck the *Automatic* toggle.
  - change the *Lens Type* to *Full Frame Fish-eye* (even if the image is actually a full frame fish-eye, this will work anyway and permit exporting to Mistika).

- use the *Align Images* option.
- You should now get a stitched image in the *Panorama Editor* window. If the stitched image appears upside down, use the *123* button in that window. Set the prompt for *Pitch field fill value* at 180 and click *Apply*.
- On the right, select *Advanced*.
- Select the *Optimizer* tab (shown only in advanced mode) and:
  - change the *Minimize lens distortion* to *Heavy + lens shift*.
  - use the *Run Optimizer*. When the prompt '*Do you want to accept the changes..?*' appears, click *OK*.
- Select the *Lens Settings* tab, and:
  - in the table *Use individual parameters for*:, double-click on the labels *Lens* and *Shift*. This should select the whole column of toggles under these options.
- Select the *Optimizer* tab again, and:
  - select the *Advanced* button (this is a different *Advanced* button from the one in the previous step).
  - in the table, double-click on the labels *a, b* and *c* to switch these columns to *Toggles Off*. We assume the lens distortion of all the lenses of the camera is very similar, so we do not want the PTGui to tweak these parameters for individual lenses. However, we know the FoV (focal length) and lens shift (imprecise centering of the lens over the sensor) can differ between lenses in the same rig, so we want PTGui to optimize these individually.
  - use the *Run Optimizer* again. When the prompt *Do you want to accept the changes..?*' appears, click *OK*.
- Use *File>Save Project* to save this alignment metadata (as a .pts file):
  - close the PTGui and go back to Mistika VR.

## 7.5 Usage of Autopano Giga from Mistika VR

### Connect Mistika VR to APG

In Mistika VR, go to *File>Options* in the main menu. In *Options*, Select *Show window to configure external Stitch applications*, so the next time you use the *Use external stitch* command, APG will be offered as an option.

## APG Configuration

Autopano Giga (APG) default settings are meant mainly for photographic panorama stitching. However, if VR Video stitching is your main use of APG, it is best to change its default settings to those typical for VR video rigs, so you will not need to do any adjustments when using APG in Mistika VR:

- Open the APG, and choose *Edit>Settings* from the top bar. A panel will open.
- In the *Images* tab, in the *EXIF* panel, activate *Always force the following EXIF value*. Below this, in *Focal*, set a value of 10.00mm. In *Lens type*, choose *Fish-eye*. Only fish-eye lens models can be imported from APG at the time of writing; this is the case for almost all VR cameras.
- In the *Detection* tab of the *Settings* window, toggle *Detection* to *Manual* and set *Quality* to *High*. Also, increase the number of control points to around 150.
- In the Optimization tab, toggle the Optimization scopes to *Manual*. For the lines marked as *Focal* and *Offset*, drag their sliders (on the right) to their extreme left (this means that focal lengths and lens offsets (centers) will be optimized separately for each lens). We recommend leaving the *Distortion* slider fully to the right, meaning lens distortion will be considered the same for all lenses, as this is usually the correct assumption.
- Optionally, you may switch the *Distortion* pulldown from *Automatic* to *Third Order*, especially for lenses with significant lens distortion around the edges of the crop circle. Do not use stitches created in previous Mistika VR versions, as these do not contain the K3 parameter necessary for the “third order” lens distortion.
- The settings will be saved and used as defaults whenever you open APG in Mistika VR or standalone. You can always reset the settings or partially override them whenever a specific case requires it.

## Usage of APG from Mistika VR

In Mistika VR, load the media. **DO NOT** apply any preset.

Use the *Stitch>External Stitch* option.

The *Stitch Configuration* window will open. Choose *Use Autopano Giga*, choose '*Do not show this window again*' and *Continue*. If you need to change your tool choice later, use the *File>Options* option and toggle *Show window to configure external Stitch applications*.

Autopano Giga will open, with a set of snapshots loaded. Above the images, use the green icon (*Detect*). APG will create a stitch and place it on the right-hand side of the APG layout. If you are happy with the quality of the stitch, in this new panel, use the pulldown menu to save the stitch file with a .pano file extension. Close the APG. Back in Mistika VR, choose the *Stitch>Import Stitch*, and choose the .pano file you just created. A stitch will be imported.

Adjust the crop circle size, unless the images were full frame (like a stock GoPro). Be aware that GUI adjusts size jointly for all lenses, which is okay, unless your rig uses a mix of different cameras or lenses (if this is the case, use the Input Camera *Crop Circle* numerical parameter to set each camera circle size separately). DO NOT adjust the crop circle positions: in Mistika, crop circle position and lens center is the same, and moving the position would break the alignment.

Some parameters can be now set:

- Increase the *Options>Feather* value to 10 to 50, which is typical.
- Activate the *Options>Optical Flow*.
- If the rig is a radial type in 3D capable resolution, set *Camera Spacing* to the correct value (60 degrees for 6 camera rigs, 45 degrees for 8 camera rigs etc).
- If the rig is based on stereoscopic pairs, go over the camera list, and for each camera, in the *Input Cameras* tab, set *Stereo Eye* to either Left or Right.

### Optional APG adjustments for individual scenes

APG, as it has been configured, should be prepared to calibrate automatically. However, some settings can be adjusted optionally before hitting the *Detect* button: above the imported images, choose the [*i*] icon to open the *Image Properties* window. Switch to the *Circular Crop* of this window and adjust crop circle sizes and centers. Centering the crop circles is critical in our case, as Mistika does not distinguish between crop circle center from lens center, and the APG optimizer will recalculate the lens centers anyway. Also, crop circle sizes can be adjusted much later, in Mistika VR, while viewing the Mistika VR stitch results. Furthermore, the focal length in the *Images* tab may require some tweaking: the optimizer uses it as a starting value, and if it is too far off, the calibration will fail. If you get an obviously failed, scrambled stitch image, tweak the optimizer with a better focal length value:

- if you can see the full image crop circle, specify 8 mm
- if you cannot see any crop circle at all (like stock GoPro), specify 16 mm
- if you can see a partially cropped circle, specify 10-12mm

- also, optionally, after the detection, and if you are skilled with APG, select *Edit* to open the full APG interface and fine-tune the stitch; add or remove control points if necessary using the CP editor and tweak the optimizer parameters etc.

*Special thanks to Pablo Ballester, APG expert user, for his help in creating the APG chapter.*

## 7.6 Creating a Clean Preset

Import the .pts or .pano project file from PTGui to Mistika VR (can be also done in Mistika Ultima and Boutique. Check for good alignment with Optical Flow switched off and feather set to a large value (30+). Align the scene to a basic position, with Camera 1 in the center (forward heading) and bake in the cameras.

In each camera, try to set the Yaw-Pitch-Roll values to round values:

- Yaw should be typically spaced evenly, for example (0, 90, 180, 270) for a rig with four cameras.
- Pitch is usually 0 for cameras pointed at the horizon.
- Roll is usually 0, +90 or -90 degrees.
- If there is camera pointing upwards, round its Pitch to +90, Yaw to 0, and find a Roll value that aligns this view, normally 0 or the angle of one of the horizon cameras.

Then:

- In the source camera, clean (delete) values of OffsetX, OffsetY
- In the source cameras, clean the OffsetX, OffsetY, and gain and hue corrections
- In *Options*, set the feather to a large value (30)
- In Ultima/Boutique, set *Compose Mode* to *Nearest Camera*
- In Mistika VR, switch *Optical Flow* off
- If you know the camera gamma curve, set it; (for GoPro, it is “GoPro flat”)

Save the stitch as a new preset (.grp) with the same name as used previously for the .pts.

## 7.7 Testing and Applying the Preset

Load the clips used for the test and apply the preset. Cameras should be at round positions and red stitch lines at neat angles.

If a part of the fish-eye crop circle is visible, activate *Show One Input* mode. Set the circle size as slightly smaller than the healthy part of the image circle. Move the circle until it leaves about the same margin of image with visible detail. Ignore bright outlines that are often produced by lens flare, making sure you take into account the circle where the scene details end.

Exit *Show One Input* mode and apply *Match Color*, then *Improve Offsets* and *Improve Angles*. Check if the scene looks acceptable without OF. Then check the scene with OF on.

## 7.8 Rig Specific Comments

### 7.8.1 Insta360 Pro

Load the media (6 movies for this camera). Then, use *Import Stitch* to load the pro.prj file or *Stitch>Use Insta360Pro calibrate*. Note that the pro.prj of one shot can be used for other shots, as long as the camera has been calibrated on an appropriate calibration-friendly scene. In our experience, calibration should stay the same for a specific camera unit for a long time. However, just in case, it may be advisable to recalibrate after a rough transport, or if results are not well-aligned anymore and you suspect that camera components may have shifted over time.

The Insta360 Pro calibration metadata is attached with each shot, within the pro.prj file next to the shot's media files. However, **do not** forget to calibrate the camera before shooting. For a good calibration, we recommend finding an open area with some details in all directions with nothing too near: a clearing in a park, or the middle of a car parking lot, for example. The resulting pro.prj should be good for all shots from the same camera unit, with the possible need to adapt it for closer scenes later using Mistika VR's Convergence parameter. To run the calibration, see the camera instructions in [How to do stitching calibration in Insta360 Pro?]

The instructions recommend stepping 1 meter away from the camera, but the calibration may be more neutral if further away or if a remote is used. The Insta360

Pro camera will apply and attach the same calibration data to all shots after the calibration until a new calibration is done.

#### 7.8.1.1 How to export the audio track with the Insta Pro 2 cameras

The Insta Pro 2 saves the audio in the file called *origin_6_lrv.mp4* file directly. If the user needs to export with the high-resolution files the audio track, the next workaround is the one to proceed to:

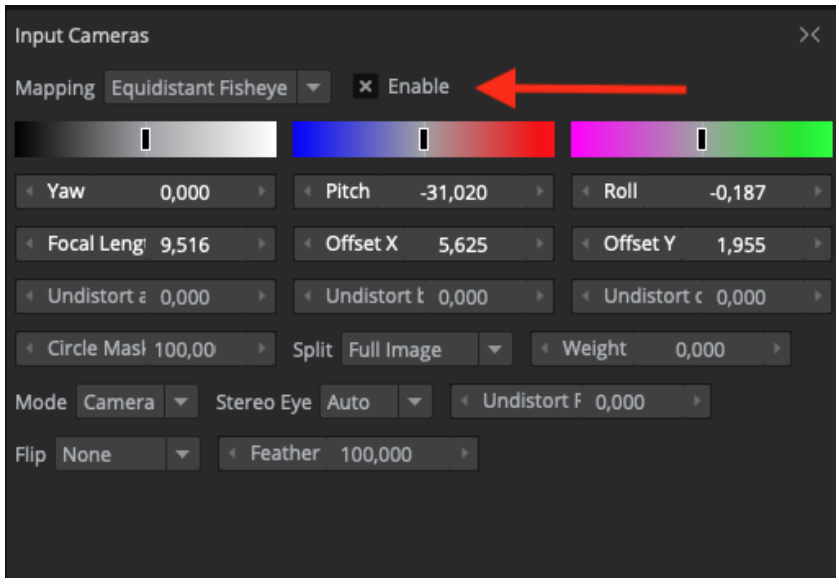
1. Import the high-resolution files.

2. Do the stitching jobs.

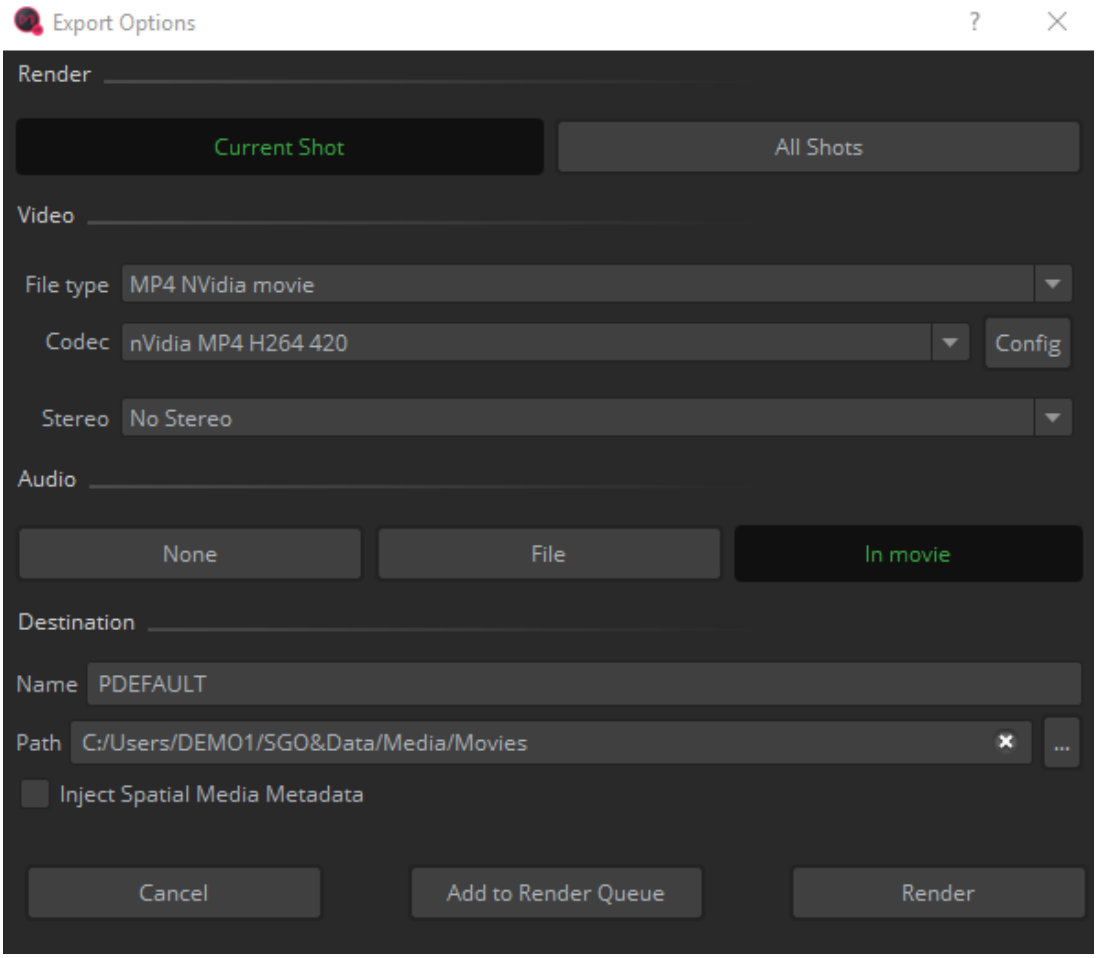
3. Once the stitching is ready, import the origin_6_lrv.mp4 file as camera number 7.

4. Select in the clip stack the origin_6_lrv.mp4 file.

5. Go to 'Input camera' control tab and do click on 'enable' to turn off the camera. As you can see the audio speaker can be selected in the clip stack so now you are ready to export your sequence with audio. Ensure the speaker in the clip stack is ON in the origin_6_lrv.mp4 file.



6. In the rendering process, do not forget to select the option for audio 'in movie'.




### 7.8.2 Insta360 Titan

There is no need of a preset for the Insta360 Titan. Once the clips have been loaded in Mistika, the .prj file include in the same folder with the clips has to be dragged and dropped to the visual editor.

### 7.8.3 KanDao

Load the media (6 movies for this camera). Then, use *Import Stitch* to load the pro.prj file or *Stitch>import Kandao calibration*.

Calibrate on one calibration-appropriate shot in the KanDao application, and save it as a project. Then import the KanDao project file: the calibration metadata from it will be applied. Some early KanDao cameras do not include calibration info directly in the camera, and do not embed it into the movies. With these movies, the *Use Kandao Calibration* option will not work. In any case, the best way to calibrate with KanDao is different: use the *Calibrate* feature in the KanDao application, on an appropriate frame of a scene suitable for calibration. Then, save the pro.prj file, and load this pro.prj file in Mistika VR. If you are using a previous version from the 8.8.8 release, first apply the corresponding KanDao preset, according to the resolution.

The calibration, if performed on an acceptable calibration scene, should be good for all scenes from the same camera unit, and seems to be quite stable over time.

### 7.8.4 KanDao QooCam 8K

The Kandao QooCam 8K camera has the particularity that all the shots are merged into onecamera clip. Therefore, to stitch them in Mistika VR, first and foremost, the two media files of the same shot have to be torn apart. However, the workflow depends whether you are working with video or photo. We'll proceed to explain it.

#### 7.8.4.1 KanDao QooCam 8K Video Stitching

1. First of all, import your media file into the storyboard. The video file will appeared as one, however, there are two media files of the same shot located in the container. We need to divide them.
2. To do so, do left click on the Clip Stack menu in order to load a preset.
3. When the Camera Presets folder is open, look for the specific folder of the Kandao Qoocam 8k for video, which is called: *"KanDaoQooCam8k_video".*
4. Now, the preset is loaded, and the two media files will be located in the Clip Stack menu.
5. Afterward, it is possible to load the Kandao calibration, through clicking *Stitch > Import Kandao Calibration.*
6. Finally, the stitching can be improved with all the differnt tools in Mistika VR.

#### 7.8.4.2 KanDao QooCam 8K Photo Stitching

1. First of all, import your media file into the storyboard. Now, import the same media file inot the Clip Stack again. The two photographs will look like as if they were the same file, however, we need to divide them to obtain the different shots.
2. To do so, do left click on the Clip Stack menu in order to load a preset.
3. When the Camera Presets folder is open, look for the specific folder of the Kandao Qoocam 8k for video, which is called: *"KanDaoQooCam8k_still".*
4. Now, the preset is loaded, and the two media files will be differentiated in the Clip Stack menu.
5. Afterward, it is possible to load the Kandao calibration, through clicking *Stitch > Import Kandao Calibration.*
6. Finally, the stitching can be improved with all the differnt tools in Mistika VR.

### 7.8.5 Nokia OZO

Import the source sequences (4 or 8 cameras), load the OZO.grp preset, and import the calibration file which the OZO software locates with its media (file ending in "*_metadata.txt".

### 7.8.6 Z Cam S1/S1 Pro calibration procedure

A key for good calibration is getting right the optical centers of the cameras. Mistika does not yet detect optical centers automatically: *Improve Alignment* only optimizes cameras' headings, rotations and shared (default) focal length. However, there is an easy way to adjust the camera centers manually, for Z Cam or for any other camera where the fish-eye outline circle is visible:

- Import all cameras.
- Select the cameras on the left, and disable all of them, one by one, clicking on the *Enable* toggle in the *Input Cameras* tab. You should be left with a blank image.
- *Select* and *Enable* one camera.

- In the graphics area, click on the center of the cross-hair to highlight the camera.
- In the *Camera Defaults* tab, reduce the *Circle Mask* until the round edges of the light blue crop limits line are only slightly larger than the visible fish-eye circle limit. About 72 for Z Cam S1.
- Click & drag on the *Offset X* parameter in the *Input Cameras* tab to move the image until its outline is approximately centered. Fine-tune by dragging with Alt key for 1/10 speed adjustment.
- Repeat for *Offset Y*. Preferably, do this calibration with a scene without extreme contrasts of brightness between sky and ground, so the extreme brightness of one side of the fish-eye circle will not give false results by blooming.
- Disable the camera and repeat for other cameras, from the point *Enable one camera* above.
- Enable only one camera, and reduce *Circle Mask* in *Camera Defaults* to a value where the circle is actually cropped (extruded in Mistika's case).

With all cameras centered, go over them again, and in the *Input Cameras* tab:

- enable the camera
- set roll to -90
- set yaw to 0, -90, -180 and -270 degrees for Cameras 1 to 4, respectively
- set the *Undistort b* value in *Camera Defaults* to -0.05 for S1; to -0.20 for S1Pro
- set the *Focal Length* in *Camera Defaults*to 8.76 for S1; 10.23 for S1Pro.

Save this as *Unoptimized preset* just in case. Now, apply *Improve alignment*.

*Improve alignment* may optimize your setting for a specific scene, but if a scene is neutral (with nothing too near, like in middle of a street), then this setting might be used as a good default, so save it as *Optimized preset*. When using the preset on a new scene from the same camera unit, first use *Improve Alignment*. Then tweak *Focal Length* in *Camera Defaults* to adjust the preferred (hero) distance at which convergence should occur. For any other distance, Optical Flow will try to produce the best results, but remember this function is not a "cure all". It is critical to try to get the geometry as good as possible first, so that Optical Flow does not need to warp too far.

### 7.8.7 Mosaic 51

Mosaic 51 camera is supported in Mistika VR, however there is no generic preset for it. To get a customized preset, including the calibration of the camera directly in the GRP file, please contact Mosaic team directly.

## 7.9 What if my camera rig model does not appear in the camera presets list?

In general, the first step for stitching is to find the best preset for your camera rig model and use it as a starting point for stitching. Mistika VR is provided with a camera presets menu including all common camera rigs on the market. But if your camera is a newer or custom-made model, then you might try the following resources:

**If available, use a camera preset for the same camera at a different resolution:**

Camera presets may also work at lower resolutions than indicated, as long as the aspect ratio is the same and the image is just a down-scaled version, not the result of sensor masking or central cropping. Some cameras use lower resolutions when increasing the frame rate, but the geometry is identical.

**You can create new camera presets with PTGui, Hugin and Autopano Giga:**

Third party applications like PTGui, Hugin or Autopano Giga permit customizing a camera rig. They provide a basic stitch with geometry information and lens distortion calibration. You can then import this stitch into Mistika VR and create a camera preset for the rig in question. Tutorials demonstrating the integration workflow for the three different products are found below:

- **PTGui**: Mistika VR - Beta Phase Tutorials 3.2 Basic Operation - Loading PTGUI.
- **Hugin**: Mistika VR - Beta Phase Tutorials 3.3 Basic Operation - Loading HUGIN.
- **Autopano Giga:** How to load an AutoPano Giga Project in Mistika VR.
- A more advanced tutorial can be found here: Mistika VR - Tutorials 4.4. Advanced Operation - Seamless Stitching.

***Note**: Please take into account that PTGui "masks" are not supported, as they are not compatible with Mistika VR edge points. PTGUI should only be used to calculate the*

*parameters related with the lens geometry and the positions and geometry transformations of the cameras.*

**Mistika VR rig presets forum:**

New rig presets are included with each new version of Mistika VR. Between versions, advanced access to new rigs can be found in Mistika VR Rig Presets.

**Ask SGO and expert users to create a camera preset for you:**

You can also ask others to create the preset for you. To do so, please create a post in the Presets Forum, or open a support ticket. You will need to provide sample images, ideally of a scene with nothing too close to the camera (a clear in a park, a parking lot, a town square etc.).

SGO


# Balancing the Convergence

# 8 Balancing the Convergence

Depending on the scene that has been calibrated, both Mistika VR, and PTGui and Autopano Giga commonly use objects at different distances to converge on in the gaps between cameras (or camera pairs for coaxial rigs). This is normally best for VR-2D. For VR-3D, convergence should normally be set at infinity, but the algorithm will frequently try to converge at something nearer. To adjust convergence:

- Switch the viewing mode to *B&W Anaglyph*
- Deactivate off *Draw Feather* (the red and green stitch lines)
- Activate *Draw Overlay* so you can see the camera centers
- For each span between adjacent cameras on the horizon, zoom into the center of that span and see if the most distant object in that direction is aligned: it should be visible in B&W, without the cyan-red horizontal fringing. If it has visible horizontal fringing, hold the Shift key, left click with the mouse at that point (horizontally near the center between the nearest cameras), and drag horizontally until the two views align and the red-magenta fringes disappear. Do not forget that there is also a span between the last camera on the right and the first camera on the left: align on either left or right edge of the image, whichever is more convenient.

In many cases, in calibration using either Mistika VR's *Improve alignment* or PTGui, the algorithm may be distracted by the fact that the floor area is usually much nearer than the horizon or the sky. It will tend to detect camera lens centers as tilted towards the floor, trying to place the floor at similar distance as, for example, trees on the horizon. Observe this by finding a vertical object at moderate distance (a person standing 5 meters away, for example) and compare its horizontal parallax (the horizontal width of the red-cyan fringing). The parallax at the feet will often be much narrower than at the head, making the object/person look top-heavy with small, distant feet. To compensate for this, use the *Vertical Balance* parameter: all the scene will apparently travel downwards, while balancing the top-bottom parallax. Usual values are negative, up to -2.0.

### When In Mistika VR

- Use *Import Stitch* to load the .pts file

- If the camera is 3D capable, set the *Options*>*cam Spacing* parameter to the equi-angular spacing between cameras forming the ring. For example: for a rig with 6 cameras forming a ring, the spacing is 360 ÷ 6 = 60 degrees.

SGO


# Optical flow parameters and advanced adjustments

# 9 Optical flow parameters and advanced adjustments

The objective of Mistika VR's Optical Flow is to identify every small feature in two overlapping images of a scene, establishing the exact pixel correspondence between them. Having this information allows Mistika VR to provide very precise stitching by rebuilding the edges of both images to make those features coincide.

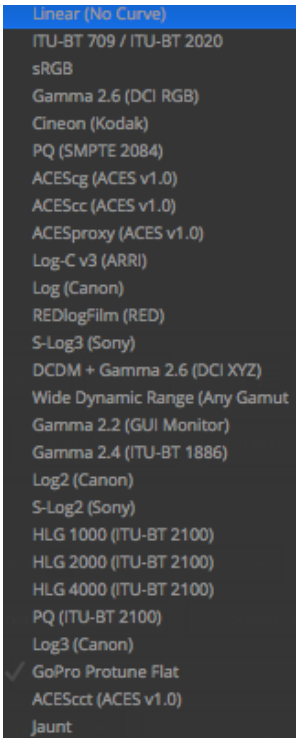
This complex calculation also involves several frames before/after the analyzed frame, tracking every small feature of the image over time and calculating motion vectors for even the smallest features of the images. We should not change the default values unless the results show noticeable artifacts. The behavior of the Optical Flow algorithm is controlled with these parameters (they appear below in the order we recommend for specifying):

## 9.1 Parameters

### 9.1.1 Gamma Curve

This is not strictly an optical flow parameter. In fact, it is found in a separate location in the options tab, but it has a lot of influence over optical flow algorithms, and should be always the first setting to check.

This parameter is also part of the source clip import settings, and will tell Mistika VR the gamma curve of the source images (check for this in your camera documentation, or, if the source images have been pre-processed, ask the originator). Setting the correct value is especially important for accurate color matching.




### 9.1.2 Range

This improves precision by telling the algorithm to concentrate on features of smaller/larger sizes (or "none" for similar priority).

Select the size that best represents the content of the stitching areas with visible artifacts. For example, if the image content is a stadium with thousands of small spectators formed by just a few pixels each, and artifacts are clearly visible on them, then selecting the *small* setting should make a significant improvement. In another example, if the artifacts are visible in areas with big buildings or other large elements, then select a larger setting.

### 9.1.3 Smoothness

This tells the algorithm if the moving features are following smooth paths or if they may experience abrupt changes in direction. This way, in cases of doubt, the function will choose motion vectors that are in parallel rather than solutions producing motion vectors that cross each other.

**Example:** Imagine two similar people wearing identical clothes and walking on a collision course. From the algorithm perspective, an action where the two people crash into each other and rebound may produce very similar images to one where

they avoid collision and simply pass behind/in front of each other. In a situation like this, the Mistika Optical Flow algorithm may become confused and choose the wrong solution, thus creating artifacts such as warped limbs from the first person sticking to the second person for a few frames (the algorithm may detect similar limbs without knowing which one belongs to which person). In these situations, you can help Mistika by telling it to expect smoother movements (people passing in front of each other, in the example) or to permit abrupt changes (people crashing into each other and rebounding, in this example).

### 9.1.4 Minlevel

This is a Resolution factor. The higher the value, the lower the resolution of the smallest features to track, thus accelerating the processing, but at the potential cost of precision. A value of 0 will track at pixel level, higher values half the resolution for each unit increment. However, take into account:

- Decreasing this parameter to 0 may result in extra accuracy or small features and fine textures, but it is only recommended for images with excellent focus, little movement and very little noise; otherwise it may be counter-productive.
- Increasing this parameter results in faster processing, but may produce artifacts.

### 9.1.5 Levels

- Refines the calculation of the direction of motion vectors.
- Increase this parameter in cases of complex movements with multiple features with crossing paths.

### 9.1.6 Iterations

This checks movement in areas of low detail. If there is not enough texture or details in the images, the algorithm may fail to track each small feature correctly. Increasing this parameter will force the function to calculate more iterations in order to find more possible solutions for the resolving equations, then compare the accuracy of each solution to find the optimal one.

SGO


# Professional features

# 10 Professional features

## 10.1 Creating 3D stitch in Mistika VR

With this professional feature, stitching is based on a preset plus a 3D Stereo tool for calibrating the rig in use and scenes shot. PTGui can be used as an alternative advanced alignment tool, as it provides a lot of control when choosing the best reference points, or when no preset is available, as in the case of experimental rigs.

Be aware that the stitch geometry adjustment should ideally not depend on the scene, so it may be helpful to create a good stitch for an easy, open scene, with nothing too near and some distant objects (such as clouds, trees or buildings) visible in most directions, and then reuse this stitch as a preset for more difficult scenes shot with the same rig unit.

- Import the clips and apply the corresponding Preset.
- Apply *Match Color*.
- Apply the *Improve Offsets* tool multiple times until no more visible changes in the resulting image are seen; 2 or 3 repetitions are usually sufficient.
- Apply *Improve Angles*, normally once, although in some cases a second application can improve results further. Multiple application will, in some cases, cause the horizon to slowly drift in a certain direction, while not actually changing the scene anymore. In this case, simply stop and undo or straighten the horizon. Improve Angles is in fact the old Improve Alignment tool, and behaves as this used to.
- Apply *Match Color* again, possibly more than once, until the results stop visibly changing.
- Activate the *Use Optical Flow* toggle.
- Switch the viewing mode from *No Stereo* to *B&W Anaglyph*.

A usable 3D anaglyph of the scene should now be visible. Anaglyph is not the best way to see 3D, but it is a good diagnostic tool as it allows the parallax of objects in the scene to be viewed.

## 10.2 Mistika VR 3D Geometry Alignment Procedure

There are basically two types of VR 360, stereoscopic capable rigs:

- **Coaxial** rigs, whose cameras are organized in multiple stereoscopic pairs. There are multiple coaxial (parallel) pairs. Examples are VRTul, Vuze, Samsung Round. VR180 rigs are a special case in this category, with a single coaxial pair.
- **Radial** rigs, whose cameras are organized in a ring, with regular angular spacing. All cameras point outward from a common center. Examples are KanDao Obsidian, Insta360 Pro, Jaunt, OZO, Z Cam S1Pro etc. Two eye back-to-back cameras are considered in this category, such as Samsung Gear, GoPro Fusion, Garmin VIRB etc.

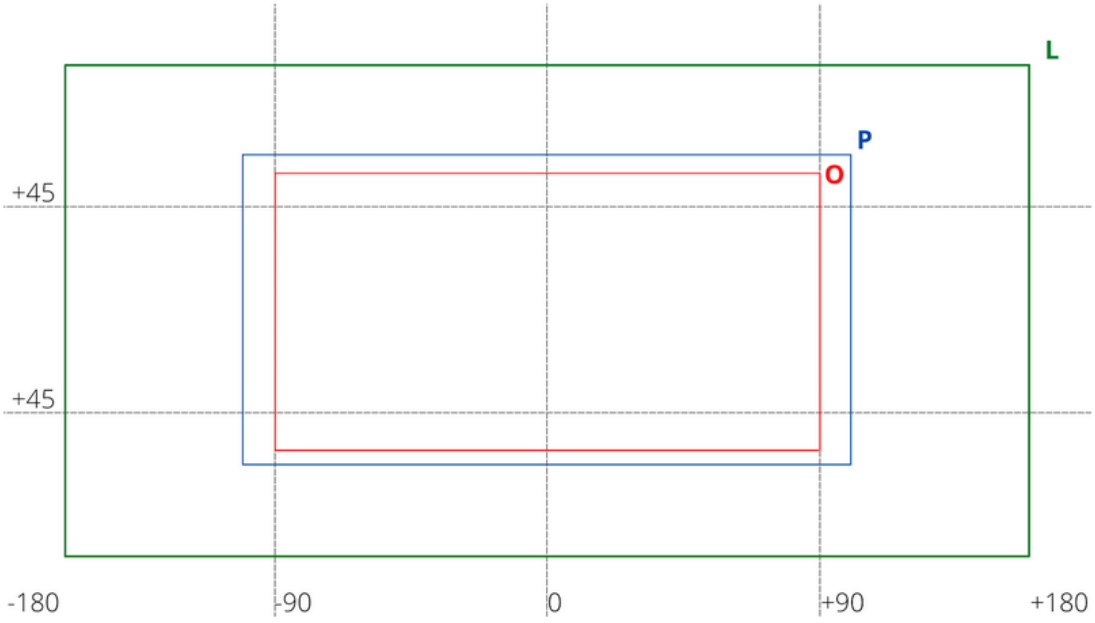
Most of the procedure is common for both types of rig. Any points applicable to one only class of rigs will be clearly indicated in this document. Both types of rig can also contain additional cameras to cover up/down directions. Find a suitable frame for calibration: if people are moving, find a frame where the distance to the nearest object (or person) is the greatest possible. If you can obtain a separate calibration shot, calibrate with this, and then use the calibrated preset on other shots from the same camera.

## 10.3 Render Options

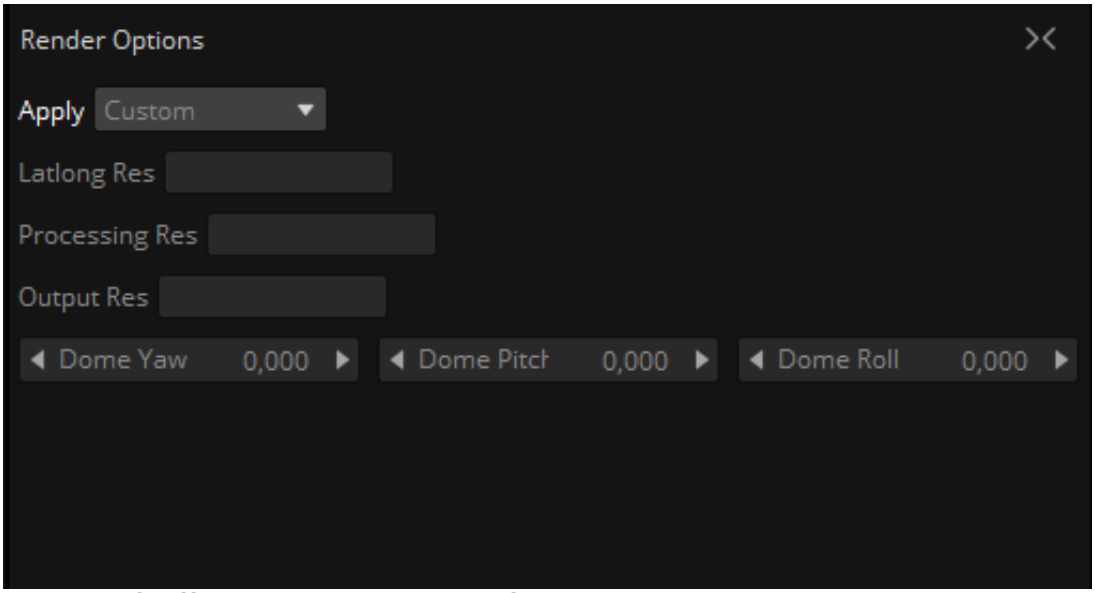
Output resolution customization depending on where our project will be projected - virtual set, dome, immersive room, etc. While working on the project, you can work at any resolution - depending also on the capabilities of your workstation. For example - you can work at a lower resolution, but then still output at the highest one - without compromising the result.

The following parameters are available to customise the cropping of the image:

- **Latlong Res** total resolution of the Latlong (entire canvas).
- **Processing Res** resolution used for processing before cropping the image.
- **Output Res** the specific clipping / part we want to render, based on the total latlong.

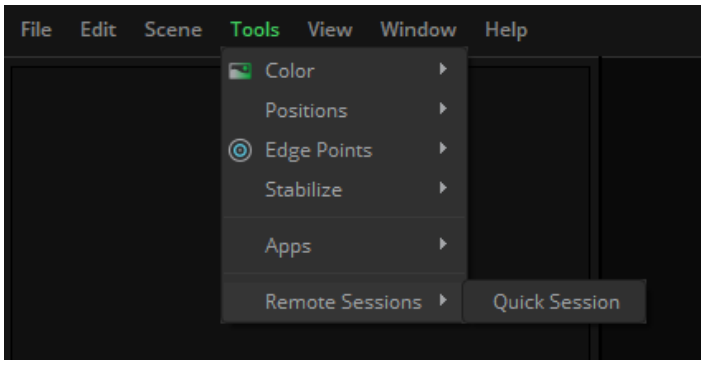



With the YAW, Pitch and Roll parameters we can now orientate the part of the image that we want to be displayed in the final render.



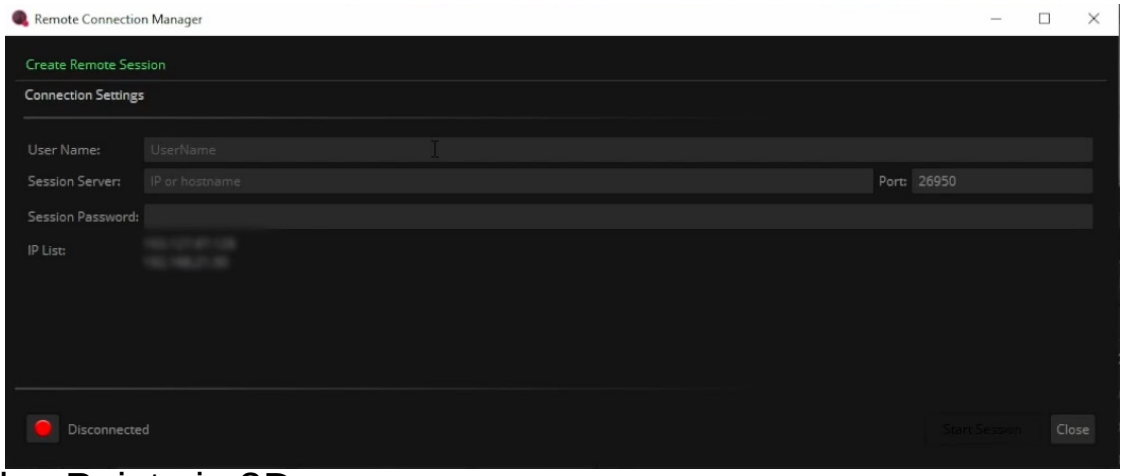

## 10.4 Remote Mistika Connect Session

To stitch your media directly in Meta Quest 2 headset using Mistika VR interface integrated. Use *Tools>Remote Sessions>Quick Session* to open the Remote Connection Manager.




Allowing you to define the default port, user-name and password. Please note that it is not necessary to use your SGO Account details. You can set up the user-name and password of your choice.

Below you will see the list of detected IP's in your system. Add your IP in the Session Server field, and when you have all the parameters filled in, start the session by clicking Connect button. When you have filled in all this information, start the session, the port, password and IP should be the same in your Mistika Connect application.




## 10.5 Edge Points in 3D

In 2D stitching, the edge points basically widen the patch around any camera, in order to push the problematic stitch lines to areas where they matter less, such as from the middle of a face to a wall next to said face.

In 3D, the basic patch, meaning a continuous zone surrounded by stitch lines, does not belong to one camera: each patch actually forms a vertical strip between two neighboring cameras. This means that the Stereo Edge Points should be added for a strip (formed by any two neighboring cameras) instead of one camera.

To add a 3D edge point, select the camera to the left of the strip (this is an arbitrary choice, as both cameras of the pair are equally important) and use the *Add Edge Point Stereo* option. A new edge point is created in middle of the strip. Now move the point to push it either left or right of the patch, just like in 2D. Use Shift+Left drag to make the Edge Point larger or smaller. Use Ctrl+Shift+Left drag to make the Edge Point Feather larger or smaller. The edge point feather size is relative to that of the overall feather, so you may not see it if the overall feather is set to 0. The Edge Points can be animated with keyframes in order to smooth moving parts of the image.

## 10.6 Stereo 3D Stitching in Mistika VR

- **Use** *Stitch>Import Stitch* to load the .pts you have just created in PTGui. You can also drag -and-drop it over the stack of cameras on the left.
- **Align** the horizon (there are different ways to do this).
- **Radial rigs only**: in the *Options* tab, set the *Camera Spacing* angle between adjacent cameras of the rig. For example: for an 8 camera rig, the angle is 45 degrees; for a 6 camera rig, it is 60 degrees.
- **Coaxial rigs only**: select the cameras individually, and for each one, in the *Input Camera* tab, switch the*Stereo Eye* parameter of each camera to either *Left* or *Right*, depending on which camera it is.
- If there are **additional cameras pointing** up/down with regard to the horizon, select these additional cameras individually, and in the *Input Camera* tab, set the *Stereo Eye* parameter to *Center*.
- Activate the Show *one input only* viewing toggle (one filled square of four).
- Activate the *Draw Overlay* viewing toggle (blue and orange dashed line circles). Use mouse- Alt+drag to reduce the overlay line circle size until it fits entirely inside the fish-eye lens cropping. Do NOT try to center the circle: it will often be offset towards the floor of the scene. DO make sure the drawn circle fits inside the lens crop circle.
- Deactivate the *Show One Input*, *Draw Overlay* and *Draw Feather* (white dashed line circle) toggles.
- Set the *Stitch Feather* to a large value, usually 100.
- Use *Color>Match Color*. Be sure you are on a calibration-friendly frame (not the first frame, in which the DP is often still touching the camera!)
- Activate the *Use Optical Flow* toggle.
- Switch the viewing mode from *No Stereo* to *B&W Anaglyph*.

A usable 3D anaglyph of the scene should now be visible, if everything has been carried out correctly. SGO will continue to refine this document as we improve the software and acquire more experience working with customers on a variety of media samples.

## 10.7 Stabilization

### Stabilize feature usage

The Stabilize function is designed to be as simple as possible to use, so it has only minimal controls. First, the one-click command is Stabilize. These are the commands to use:

The Stabilize pulldown contains following entries:

Stabilize: this is the one-click command. It will:

- Bake in the Output Camera setting to clear it for locating the stabilization information.
- Scan the range defined by the time marks for overall scene movement.
- Apply the stabilization according to the “follow” setting.

During the Stabilization scan, it is good idea to deactivate *Optical Flow*. It usually makes little difference, and will slow down the scan.

- *Follow Overall Heading*: if this toggle is activated, the stabilization will try to compensate only camera shake, and will respect the overall heading of the rig, so Camera 1 will generally stay centered. This permits stabilizing the camera when moving a curved route, like a car driving around. If the toggle is deactivated, the stabilization will try to keep the view as steady as possible, but it may become confused by one side of the scene advancing while does other not, making the scene gradually roll or tilt.
- *Follow heading after x frames*: this is the number of frames the stabilization will settle on a new camera heading. A low value means swifter following, but possibly with less smooth results. Higher values mean slower reorientation time. If stabilization seems to drift too far from the intended orientation, reduce the value from 500 to 100, for example.

- *Reapply stabilize*: this will reapply the last stabilization applied, using a new setting for the *Follow* toggle and value, without needing a rescan.

The stabilization is applied in 3 new parameters in the Output Camera: S-Yaw, S-Pitch and S-Roll. If the stabilization is to be deleted, reset these 3 values by right-clicking over them one by one and choose *Default Value*. You can also reset the whole output camera by right-clicking in the panel but not over a value, and use *Reset all values in this panel*.

Be aware that the stabilization process analyzes the 'equator belt', from -45 to +45 degrees of latitude, for movement. It avoids the polar areas, as these are usually the least useful for tracking. This means that if your scene is strongly tilted in the raw shot, you should reorient it to an approximately horizontal position before starting stabilization. After stabilization, you can reorient the output camera as usual, and the reorientation will be applied on top of the stabilization.

### How to use The Stabilization

Set the render marks to delimit the stabilization time range. Then approximately level the horizon on a frame representative of the range to stabilize, so the stabilizer obtains a reasonably leveled horizon to start with. Run the Stabilize command, and then level the horizon again, and it will be applied on top of the stabilization. During the Stabilization scan, it is a good idea to switch off Optical Flow. It usually makes little difference and slows down the scan. Stabilization normally should be run after aligning the scene and matching colors.

## 10.7.1 Gyroscope (IMU) based stabilization

(Courtesy from Insta 360

Gyroscope metadata integration support allows Mistika VR users to easily import the motion sensor metadata from the supported cameras. This feature provides stabilization based on the physical information of the shot, providing better results regardless of the lighting conditions of the scene (this is especially useful for flashy and low-light conditions footage, such as crowded concerts for example).

Many modern cameras include a Gyroscope (or IMU - Inertial Motion Unit) sensor that records the exact orientation changes of the camera along each moment of the

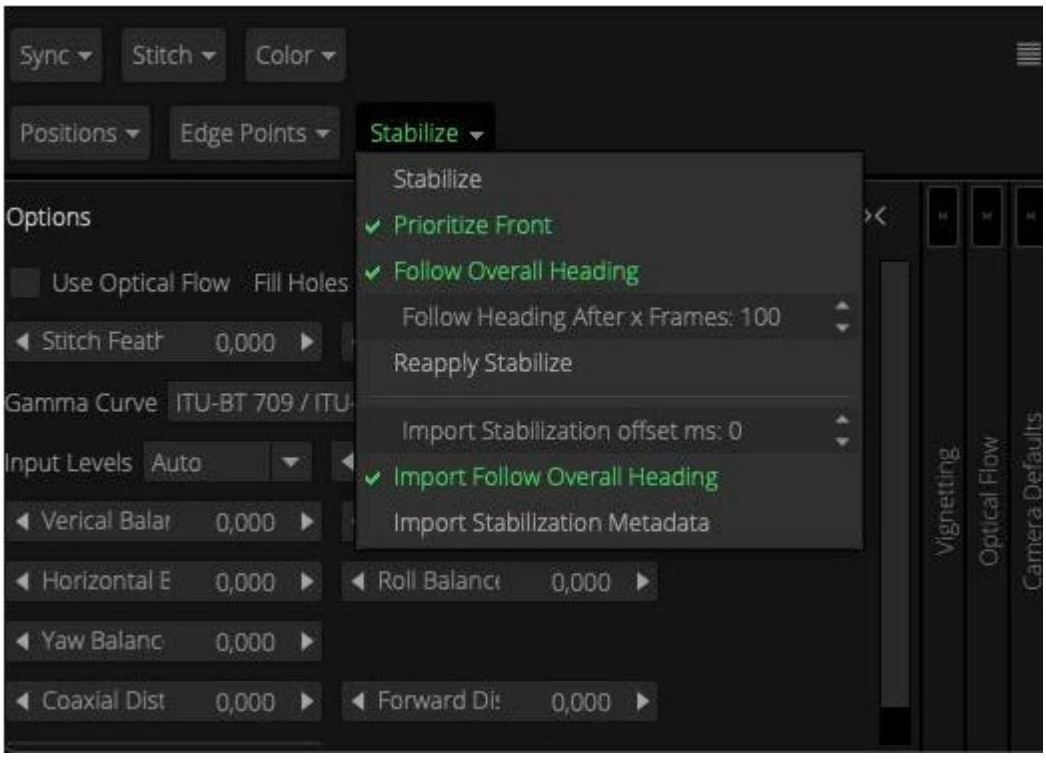
shot. This orientation metadata is recorded along with the video, and allows to stabilize the shot during the optical flow stitching.

At the moment, Mistika VR supports IMU stabilization for the following cameras:

- Insta360 Pro2 and Titan (under their proprietary name, Flowstate)
- KanDao Obsidian and QooCam8k
- Teche 360Anywhere

**How to apply Gyroscope Metadata?**

1. Import the media and stitch the footage by applying camera preset or importing calibration.
2. Import the stabilization metadata by clicking on Stabilize > Import Stabilize Metadata.




After clicking on the button, the stabilization is imported and the distorted horizon is now steady, as in this screenshot:


*Thanks to Hugh Hou for providing the footage.*

Now, the S-Yaw, S-Pitch and S-Roll parameters are animated (their values are blued). They bring the stabilization information and their values are modified to stabilize the image.

To fix the possible time misalignment, we recommend following this procedure:

- Find a short segment of 1 or 2 seconds where there is a regular rocking movement with the overall camera direction not being changed (the person carrying the camera is walking in one direction, the car is not making any turns).

- Then place the yellow play marks on the limits of the segment and click the play button. The segment will now play in a loop. Do not activate the optical flow parameter as we need a good playback speed.

- Find a part of the scene, in forward direction, which moves only because of camera wobble, and does not advance across the screen. Zoom in on this part, center it using the right mouse and drag. If the play stops, use the play button again.

- With image playing in a loop, change the “import stabilization offset” parameter to +5ms, and apply the “Import Stabilization Metadata” again. The play loop should now continue, and you can see if the stabilization has improved.

- If +5ms makes the stabilization clearly worse, apply -5ms value instead. You can try different options until you find a value that works best for your shot.

## 10.8 Keyframing

Any Parameter can now be animated. To control the animation, open the contextual menu for the parameter by right clicking on it. There will be these new options:

- *Default value*: totally resets the parameter. If it was animated, animation will be disabled.
- *Add a keyframe***:** a keyframe will be added at the current frame position. Animation is enabled for the parameter, if it wasn't enabled already. From now on, any change to this parameter will automatically insert new keyframe in current time, if there wasn't a keyframe there already.
- *Remove a keyframe*: a keyframe at the current time will be removed. If it was the last keyframe left, the curve will become non-animated.
- *Remove animation***:** animation will be disabled, all keyframes removed, but the current value of the parameter will be kept as a non-animated value.

Keyframes for selected parameters are shown as green marks, while animated segments will be drawn as light blue segments, corresponding to the color prompts above. The numerical value of the parameters is color coded:

- **Gray numbers** show the default, unmodified value.
- **White numbers** show the value was set by user. It may be set to the default value and will still show white as it was set by the user. To completely remove the value set, use the “default value” command.
- **Green numbers** show a keyframe value set by the user.
- **Light blue numbers** show a value interpolated between keyframes.

## 10.9 The Alignment Mode Tool

This experimental tool assists in fixing vertical parallax issues. While in theory it may work for many types of rigs, it is only simple to use for VR180 parallel pair rigs. With this tool, you can selectively drag vertically parts of one camera: Center, Left and Right. Mistika VR will modify a combination of camera parameters, *Yaw*, *Pitch*, *Roll*, *OffsetX* and *OffsetY* to move only the selected zone, while moving the other two zones very little.

- First, always center the crop circles for both cameras the best you can, as these are the main source of misalignment, and are a basis for good results.
- Enter the *B&W anaglyph* mode to see the vertical misalignments.
- Preferably choose a frame where there is no object to camera in the exact center.
- To enter or exit the Alignment Mode, toggle the *AM* button in the toolbar, found next to the overlay modes.
- Be aware that this mode will only work correctly if the output camera is unmodified. If the horizon was aligned, use *Bake in output camera* first.
- Select the camera to be moved by Shift+Click on its center. If the two centers overlap to much, select one of the cameras in the camera list on the left.
- Move the camera center into alignment: in *B&W anaglyph*, the image should become black and white, without any color fringing. Only for the center, you will be able to align in both axis (*yaw* and *pitch*).
- Follow the horizon line to one side (left or right) and drag the mouse up/down until it aligns vertically. If possible, it is best to look for horizontal objects or clearly defined objects which you can easily see are at the same height or not. Try to align on objects that are nearest to the horizon, because for objects above or below it can be misaligned given the phenomenon of 'interaxial distance'. Some horizontal parallax will usually remain with colored fringing, since the sides are not necessarily at the same distance as the center.
- Do the same for the other side (*left* or *right*).
- You might need to recheck and realign the center and both sides again: while the tool tries to move the zones as separately as possible, this is not yet a perfect process.

## 10.10 Autocalibrate

Based on global stitching optimization, capable of calculating lens distortion and focal lengths of individual cameras, in addition of their sensor offsets and heading angles whose calibration was already supported by the "improve offsets/angles" features. The autocalibrate feature is specially beneficial for 3D stitching, as it produces much better vertical alignment of the stereoscopic view.

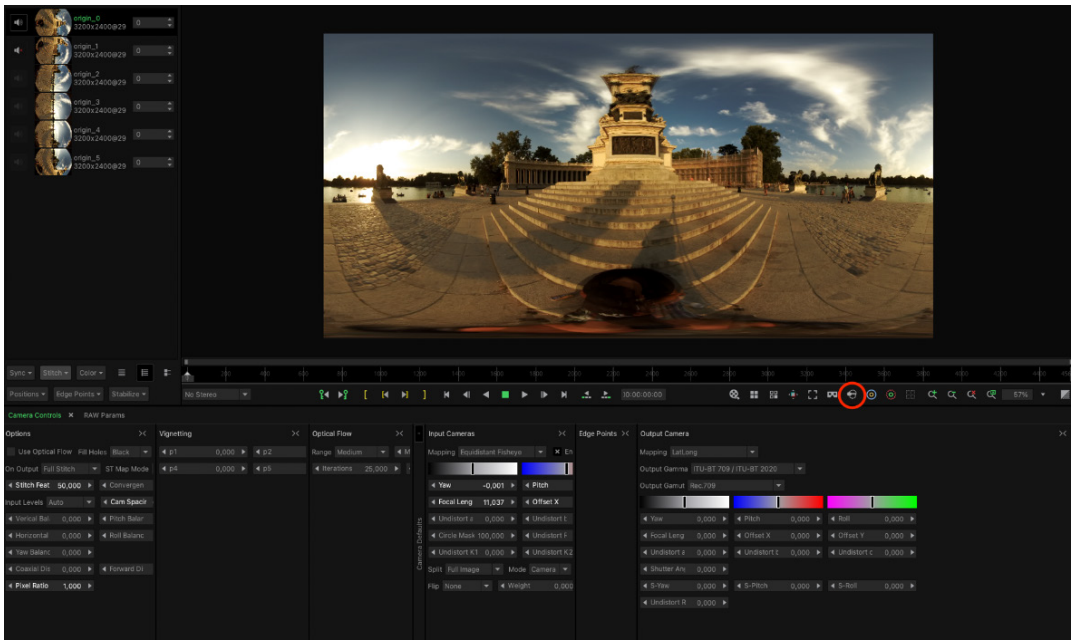
## 10.11 VR Headset preview

This functionality enables a live output of your stitched VR media to VR Headsets without the need of rendering. This functionality is being supported through the integration of DeoVR, a platform-agnostic and completely free of charge VR player.

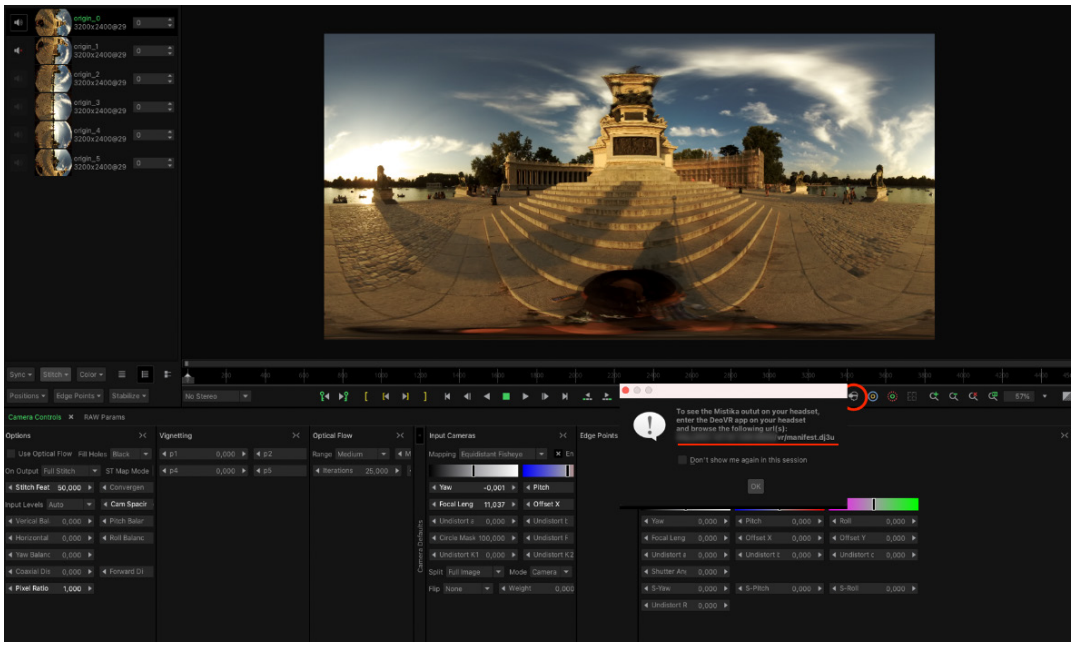
The signal will be sent through the internet connection. Please make sure to use the same network connection for both - the workstation where you run the Mistika VR on and the VR Headset where you want to see the preview.

**How to enable the VR Headset Preview?**
https://www.youtube.com/watch?v=eaK33JHSgkE

To activate this feature, click on the  icon you can find among the controls of the Storyboard:



When you click on it, a pop up window will appear, instructing you to copy your computer IP website address and paste it in the DeoVR player:



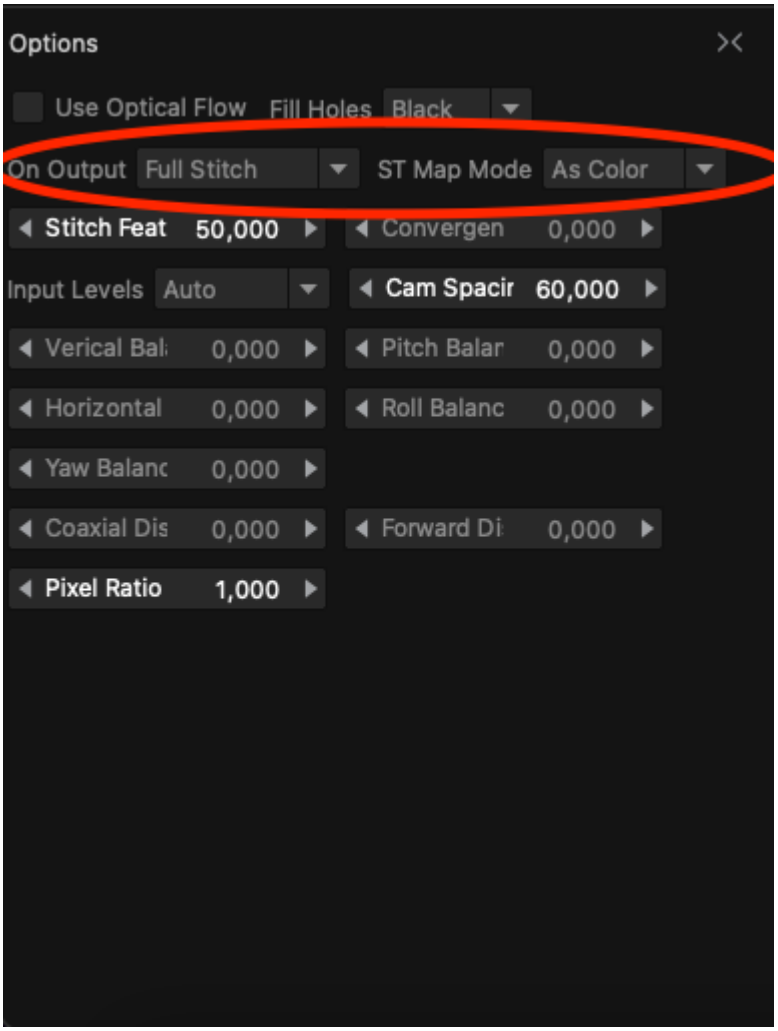
Open the DeoVR player on your preferred platform and paste the IP address. If you do not have the DeoVR player installed yet, click **here** and install the application.

Make sure that you click the play button in Mistika to get the video output in DeoVR.

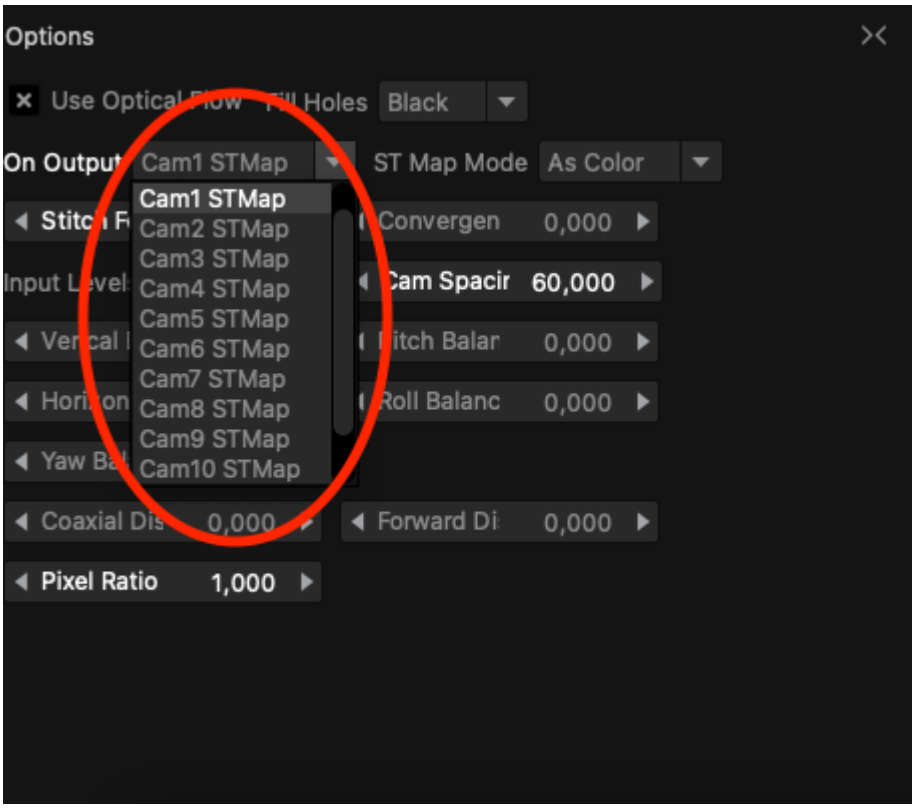
## 10.12 STMap Integration

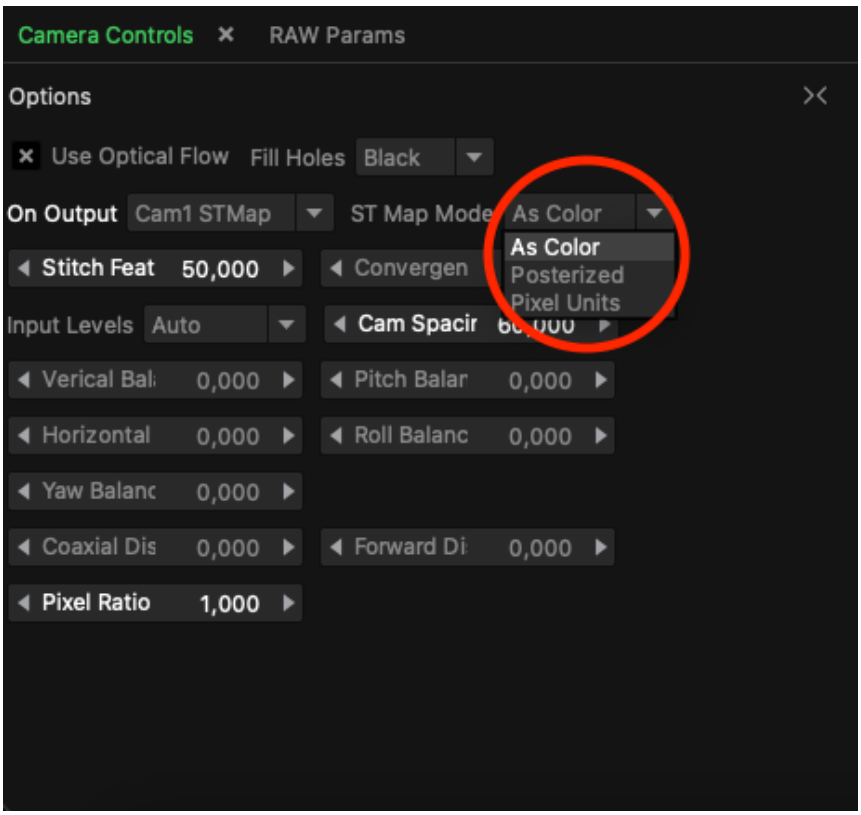
Facilitating the workflow between Mistika VR and compositing softwares such as Nuke, STMaps provide all the stitch information, not only geometry but also optical data.

In the options tab, you can find the options to export STMaps:



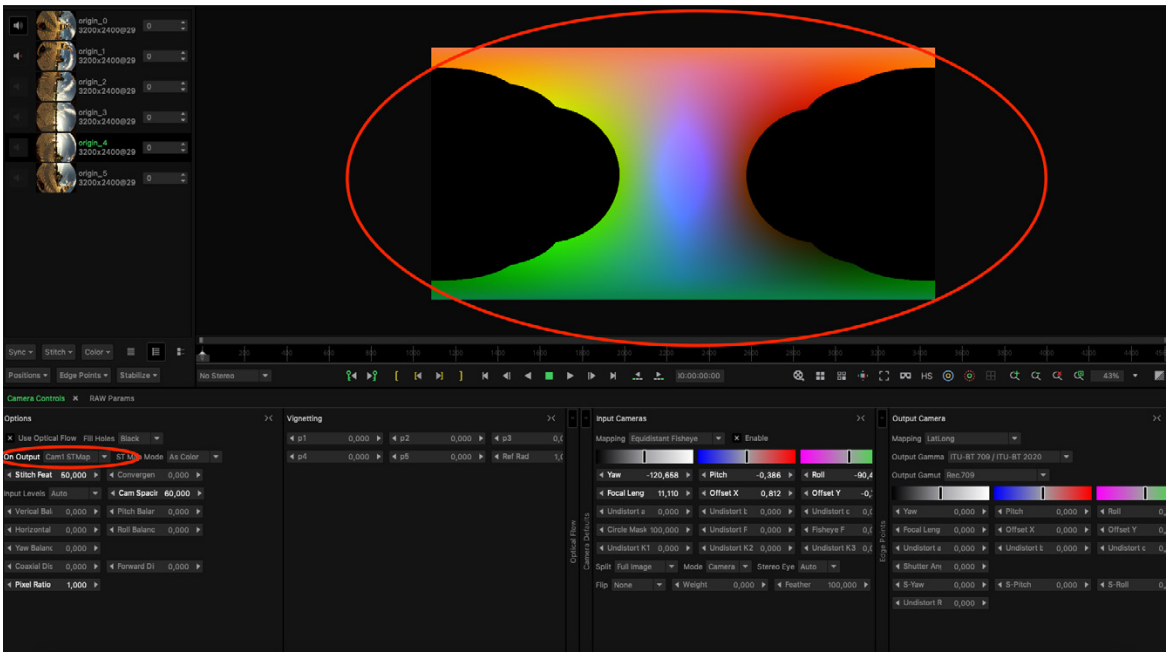

- **On Output** allows you to visualize the different ST Maps of each camera. The "Full stitch" option is the default mode. It turns off the ST Maps and displays the stitch.
- **ST Map Mode** displays the different ways to visualise the ST Maps (As color, posterized or pixel units).




Please, remember the Optical Flow has to be activated to be able to display the ST Maps.

The chosen option in "On Output" parameter will be your render result.


origin_0
3200x2400@29
origin_1
3200x2400@29
origin_2
3200x2400@29
origin_3
3200x2400@29
origin_4
3200x2400@29
origin_5
3200x2400@29
Sync
Stitch
Color
Positions
Edge Points
Stabilize
No Stereo
Camera Controls
RAW Params
Options
Use Optical Flow
Fill Holes
Black
On Output
Mode
As Color
Stitch Feat
50,000
Convergen
Input Levels
Auto
Cam Spacir
60,000
Verical Bal
Pitch Balar
Horizontal
Roll Balanc
Yaw Balanc
Coaxial Dis
Forward Di
Pixel Ratio
1,000
Vignetting
Input Cameras
Mapping
Equidistant Fisheye
Enable
Yaw
-120,658
Pitch
-0,386
Roll
Focal Leng
11,110
Offset X
0,812
Offset Y
Circle Mask
100,000
Fisheye F
Split
Full Image
Mode
Camera
Stereo Eye
Auto
Flip
None
Weight
Feather
100,000
Output Camera
LatLong
Output Gamut
Rec.709

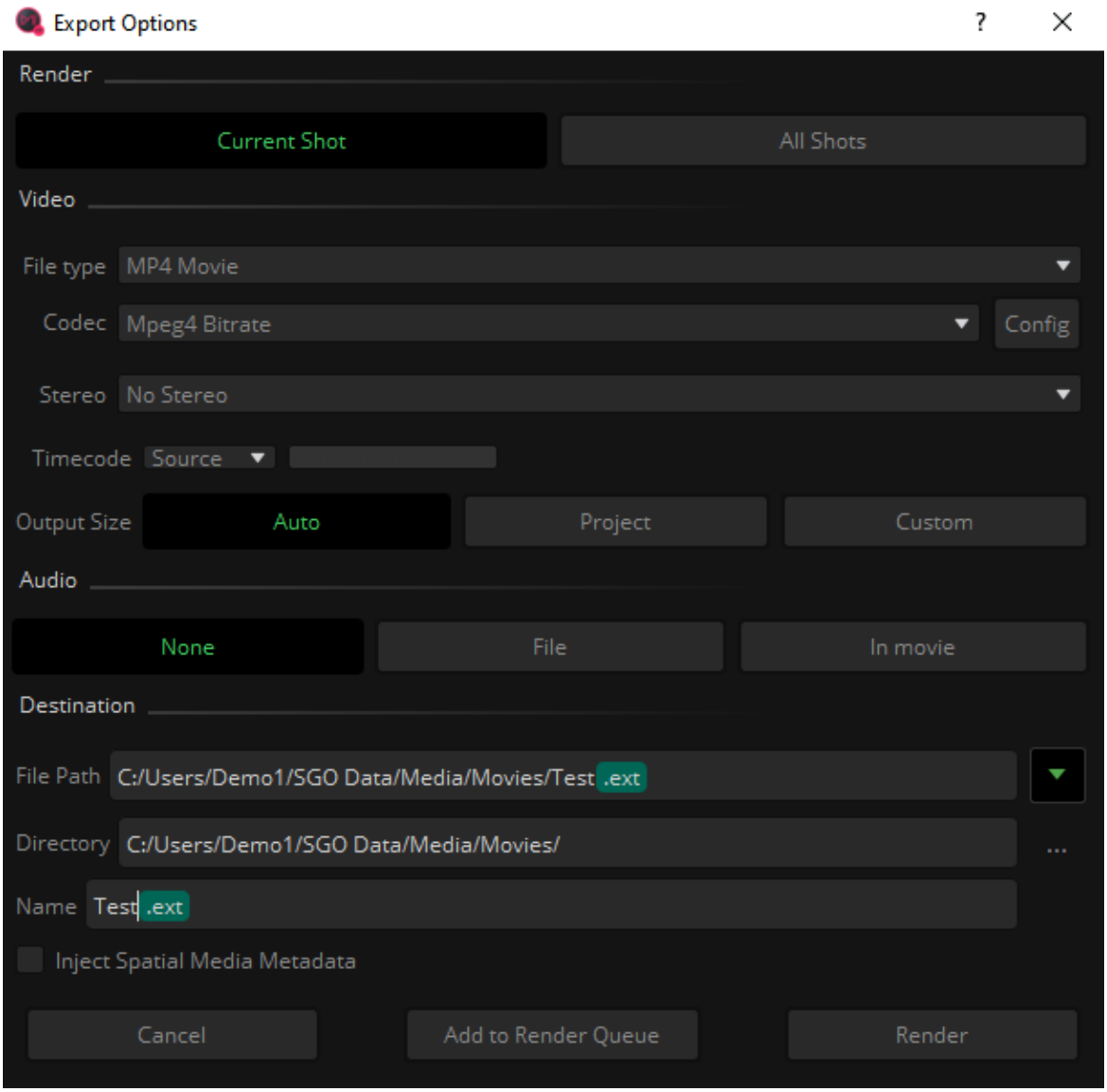
SGO


# How to render and formats in Mistika VR

# 11 How to render and formats in Mistika VR

## 11.1 How to render in Mistika VR

The Export Options menu is located in *File* > *Render:*




First, it can be decided whether to render only the *Current Shot* opened in the Storyboard, or render *All Shots* with the same export settings. All the shot will be rendered in case there are not In and Out marks which shortens the length of the rendered shot.

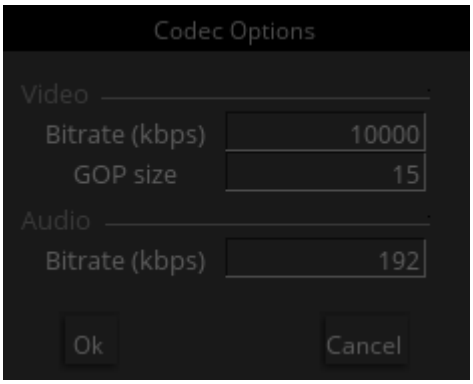
Secondly, it has to be decided the file type between these options:

- MP4 movie
- MP4 Nvidia movie (available for Nvidia graphics cards users)

- Mistika
- Image sequence
- Quicktime ProRes

Each of the files has a different codec to decide whether the workflow or the media demands high or low compression codecs. The following chapters explain in detail the specifications of each render file and their corresponding codecs

The MP4 codecs can be configured with different bitrate and gop size options if Config button is selected, which opens the Codec Option menu:




The Stereo menu allow to export all the different Stereo display options:

- Top and Bottom
- Left/Right
- Left/Right VR180
- Left view
- Right view
- Anaglyph
- B&W Anaglyph

Timecode can be exported in different ways:

- Source: Render will include the TC of the imported clip.
- Custom: The TC can be determined by the user.

The Audio tab allows to decide to export non audio at all, to export it as a different *File* in the render path selected, or *In movie*, which injects the audio in the same rendered file.

The Destination tab allows to write the render Name and choose the destination Path. By default the render Path is the following: *Users/*Username*/SGO Data/Media//Movies.* The *Inject Spacial Media Metadata* checkbox (not available in all render files) activated the VR view in the export for third party players that are compatible with 360 metadata. Finally, the Render button activates the whole process, which shows up a loading bar window that indicates the evolution of the render until it is finished.

**Media Output naming**

The Destination tab allows you to write the render Name and choose the destination Path. By default the render Path is the following: *Users/*Username*/SGOData/Media//Movies.* However, it is possible to give the original source name to the media or select other options to the render name.

You can do so by using the Dynamic tool, enabling tag auto-fill in name convention. This functionality allows to select several tokens, such as resolution, color space or framerate and they'll be part of the destination path or render name.

These tokens can be added by using square brackets [example]. When you create the first bracket you get a drop-down menu with the different tokens, write or select the one you want and the bracket will close automatically.

When you start typing your chosen token in the Name Convention the search engine will automatically recognize and filter out the most appropriate tags. For example: if you write "[f" the drop-down menu will show: fStops, FocalLenght. fps, frame

**The Inject Spacial Media Metadata checkbox (not available in all render files) activated the VR view in the export for third party players that are compatible with 360 metadata. Finally, the Render button activates the whole process, which shows up a loading bar window that indicates the evolution of the render until it is finished.**

## 11.2 Render Formats

Many codecs are relatively slow compared to Mistika VR's render speed, so they may cause bottlenecks in overall render performance. The best options for intermediate renders, to be passed on to next stage of the post-production chain, are typically:

- **mp4 NVIDIA Movie**: the codecs are extremely fast, but have some limitations:
  - o Resolution for H264 is limited to 4Kx4K
  - o Resolution for HEVC is limited to 8Kx8K, or 4Kx4K for older nVidia GPUs
  - o The *QP* modes are suitable as intermediate formats, with consistent quality adjustable from lossless (QP=0) to visually lossless (QP=1..5), high quality (QP around 10) normal quality (QP around 20) all the way down to heavilly compressed (QP=50). Use low GOP values for intermediate formats for better scrubbing performance in other apps. Options of this mode are *GOP* and *QP*.
  - o The bitrate formats allow maximum bitrate of 135.000 Kb/s. This is quite low for a 6K image, for example, and can cyclically produce drops in quality with each GOP. Use only if defining a specific bitrate is important. Options of this mode are *GOP* and *Bitrate*.
- **ProRes**: this format is an excellent intermediate, but requires a lot of CPU power. The codec speed scales very well with the number of CPU cores in the system.
- **DPX** sequence: this format is the maximum quality standard, lossless 10 bit. It is very fast, limited only by the speed of the storage in use.
- **EXR** sequence: this is an image format developed by ILM and mainly used as an intermediate format for VFX workflows. Probably the most popular variant is DWA (created by DreamWorks Animation). It is an image sequence (one frame per file), which makes it ideal for many VFX workflows, and in fact most VFX applications support this format.

Please note that only the most common codecs are mentioned here, and some technical explanations are simplifications, since this document is not intended for advanced users. Mistika VR currently has render support for the following tracks:

- **1 Video track**, also with support for Stereo 3D in split image modes.

- **2 Audio channels**: simply mute the cameras that should not be audio sources. The user can choose between audio embedded in a movie file or as a separate file.

- The rest of this chapter is divided in two sections, **Compressed codecs** and **Uncompressed codecs**.

## 11.3 Compressed codec

Compressed codecs produce small files requiring low bandwidth, but some information is lost on each generation (although there are some lossless variants). In general, they are best for final deliveries, and also for images that will not suffer more changes in post-production, but only lossless variants (explained below) should be used for transferring images to other post-production applications if they are intended for anything other than the most basic editing tasks.

Please note that most cameras perform a first generation compression, and final consumer deliveries will also carry out another. Adding more compression stages at intermediate renders rapidly degenerates image quality.

*NVidia hardware codecs* vs. *software codecs*: NVidia codecs can render much faster than software codecs (typically 10x faster), while producing similar quality to their software equivalents. However, they require some resources that may not be available:

- **Latest generation NVidia board**: maximum resolution depending on each model. Click here to find the available render resolutions for each GPU.
- **Windows or Linux OS**: (Apple does not support NVidia encoder at the time of writing).

The render speed of NVidia codecs depends solely on the GPU, while the speed of all other codecs depends solely on the CPUs (except in the case of uncompressed codecs, which mainly depend on storage speed).

- **444 vs. 420**: simply expressed, YUV 420 means that part of the chroma information is shared between adjacent lines of the image, while RGB 444 keeps all the sensor information. 420 produce smaller files with no significant visual impact, but it removes information that can be important for chroma keys and other VFX tasks. As a result, 420 is an excellent format for previews and for final delivery to end consumers, while RGB is more appropriate for

intermediate renders and VFX. However, this also depends on the quality of the original cameras: if the originating cameras already compress to 420, then maintain this format; check your camera specifications for this information.

- **H264 vs. H265:** H265 provides higher quality at the same bitrate, and also supports higher resolutions up to 8K (H264 only supports levels up to 4K). However, H265 is slower decoding and playing back than H264.

For these reasons, we generally recommend H264 for HD & 4K, and H265 for 6K & 8K.

- **GOP factor:** Many codecs have a definable GOP (*Group Of Pictures*) value, found in their settings wheel. The GOP essentially defines the number of frames between keyframes.

Simply expressed, keyframe compression does not depend on adjacent frames (think about it as a .jpg image), while intermediate frames between keyframe only contain their 'differences' with the next (or previous) keyframes. High GOP values produce smaller files and reduced bandwidth needs for streaming applications, but they are not recommended for editing applications, as they will need a lot of processing to jump to a particular frame (and some applications do not even support GOP editing at all). Keep it low for editing and VFX, and increase it for previews and for final deliveries. The most appropriate GOP also depends on the content, so more specific rules cannot be provided here.

- **Lossless compression vs. QP**: if the rendered images are going to pass through professional color grading or high end VFX processing, then use *Lossless* compression whenever possible.

Simply expressed, *Lossless* codecs don´t change the images, or at best only affect information that was already lost due to the cameras' signal-to-noise ratio, but that is enough to reduce the file size significantly (typically 3:1 or more). *QP*, on the other hand, permits user defined compression by setting the *Quantization Parameter* (lower value = larger and higher quality). Please note that when using *QP*, subjective 'quality' is constant, not the bitrate (the bitrate will constantly evolve using more or less bitrate depending on the content, in order to achieve a target quality factor).

As a result, the interpretation of the *QP* value is content dependent, so there are no exact rules to follow. THat said, for average content, values of 10 and below should produce high quality lossless results, ideal for intermediate renders for transferring results to other post-production applications. In the other direction,

values between 11 and 24 are appropriate for final consumer deliveries providing good visual quality (although not enough data for applying further post-production tasks) and even higher *QP* values, 25 or more, will produce small files of low quality containing noticeable artifacts.

- **QP compression vs. Bitrate:**

The *QP* modes are suitable as an intermediate format, with consistent quality adjustable from technically lossless (QP = 0) to visually lossless (QP = 10) and quality lost beyond that point. In other words, high quality (QP around 10), normal quality (QP around 20) down to strongly compressed (QP = 50). Use low GOP values for intermediate formats for better scrubbing performance in other applications, and higher GOP values to produce smaller files at the cost of editing interactivity (and potential quality issues at some point). Options of this mode are: *GOP* and *QP factor*.

- **Bitrate:**

The *Bitrate* modes allow a maximum bitrate of 135.000 kb/s. This is too low for a 6K image, for example, and can cyclically produce drops in quality with each GOP (good quality on keyframes, then progressively lower quality between them). Use *Bitrate* mode only if defining a specific bitrate is important (and even with that, it is not very precise and considered obsolete for modern formats); otherwise, it is always better to use *QP*. Options of the *Bitrate* mode are: *GOP* and *Bitrate*.

- **Inject spatial metadata:**

This setting adds some extra metadata to tell the playback applications that the media is an VR 360 file. If not present, general purpose players may fail to switch to 360 mode, and play the images as they are, even if they support 360.

- **Main lossless codecs:**

  - **NVidia HEVC 444 10b Lossless**: no useful information is lost, and it provides significant file size reduction compared to uncompressed codecs. However, they can be slow to decode, and not all applications support HEVC.
  - **NVidia HEVC 420 8b Lossless**: produce smaller size, but being 420 8bit means that it is only lossless for low-cost cameras that are also based on YUV420.

- **Main general purpose codecs:**

  - **NVidia mp4 H264/H265(HEVC) QP**: this is probably the best codec for general purpose deliveries. It provides the best compression and speed at the same bitrate.
  - **H264:** provides great compatibility and it is fast to decode, ideal for realtime playbacks up to 4K. If you are not sure about what codec to use, use this one as a default
  - **H265/HEVC:** supports 8K and provides additional quality at the same bitrate, but it can be much slower to decode.
  - **Apple Prores**: this is the standard family of codecs for Mac computers. In general, it provides similar quality and size as the equivalent H264 variants. Official support for the latest Apple ProRes SDK is provided in the latest release. It provides high improvements in both playback and render performance, and more accurate metadata management. You can find all the information related to Apple ProRes codecs support in their official web.
  - **JPG 8b image sequence**: since this is an image sequence, it is not so efficient in terms of compression because it can not compress across time, and it is also slower to render. that said, it still it has a few advantages: it is very compatible, even with old applications; and unlike movie files, it can be rendered in parallel (several render nodes working together to render the same clip) with render managers like Smedge, Deadline, or Mistika Ultima BatchManager; it can also be partially 'patched' when needed (unlike movie files which need to be completely rewritten).

*Note: In QP formats, Quality is user-defined via the Quantization Parameter (lower values mean better quality & larger files).*

## 11.4 Uncompressed codec

These formats do not apply any compression except for their nominal bit depth and sampling limitations. They are ideal for sending media to high-end VFX and other professional post-production tasks. In general, they are larger than lossless codecs, but also more compatible and faster to decode, as long as the storage in use is fast enough, as they require large and fast storage volumes.

**Uncompressed Movie Formats:**

- **Mistika .js**: this is the only uncompressed Movie format available (all images of a clip contained in the same file). It is designed for perfect memory alignment and optimal realtime playback performance (up to 8K 60p uncompressed in realtime). It also supports parallel render with the Mistika Ultima Totem tool, and avoids file system fragmentation problems. This format is only supported on Mistika applications, but if the user plans to work on such applications, it is generally the most optimal format.
- **Uncompressed Image Sequences**: apart from maintaining the best quality, because these are enumerated sequences, they can be rendered in parallel with Smedge, Deadline, or Mistika Ultima BatchManager. They can also be patched by parts:
  - **DPX RGB 10b**: this is probably the most standard format in the VFX industry and does not have resolution limits in itself.
  - **Tiff 16b**: this is used to produce masters deriving from very high-end cameras with a great deal of extra HDR range. It produce extremely large files offering no real benefit in most cases. Only use this format if you are absolutely sure that you need it.
  - **EXR DWA:** this is an image format developed by ILM and mainly used as an intermediate format for VFX workflows. Probably the most popular variant is DWA (created by DreamWorks Animation). It is an image sequence (one frame per file), which makes it ideal for many VFX workflows, and in fact most VFX applications support this format.A difference to movie files it can be rendered in parallel (several render nodes working together to render the same clip) with render managers like Smedge, Deadline, or Mistika Ultima BatchManager. Also it can be partially "patched" when needed (as a difference, "movie" files need to be rewritten completely).

    This permits an interesting workflow for Mistika VR users: You can render a first stitch for initial review (for example with Mistika Boutique or Mistika Review products) and then adjust and re-render only the parts containing fixes. To do that just put the edit marks in the area to fix and render with the same name, so Mistika will only overwrite the selected frames . (obviously this technique only works as long as you do not need to move the horizon or other geometry settings anymore).

    Another advantage of the EXR format is that it supports 16bit per channel, which makes it ideal for HDR workflows and for high end cameras in general. EXR DWA provides definable compression, with the default compression value (45) it is considered a "Lossless" format, even for high end cameras.

The EXR DWA format is CPU intensive, but as a difference to jpg this format is fast to encode and to playback, being possible to playback in Mistika applications at 4K when having decent CPUs. Another EXR variant of common usage is EXR ZIP. This is made by compressing the original image to ZIP format internally ( the ZIP compression is not only lossless but also fully reversible). But It is much slower and less efficient in terms of file sizes when compared to DWA, so it is only recommended when explicitly required by an VFX department.

SGO

# Troubleshooting

# 12 Troubleshooting

SGO aim to constantly update and upgrade this section of this manual with the valuable input of the VR Community. Please feel free to share your favorite tips & tricks regarding Mistika VR by sending a description of the problem and step-by-step instructions to resolve it to either of the following addresses:

marketing@sgo.es *or* support+vr@sgo.es

- **Artifacts on edges of overlapping zones:** try setting the *Range* parameter in *Optical Flow* to *Large* or *Medium* to see if stitching improves. Larger settings mean larger displacements will be stitched, but the stitch can more easily “fall off” the edges of each camera view and produce artifacts in the areas of the stitches.
- **Black hole in the zenith or nadir:** lack of coverage or the imperfect correction of lens distortion can leave a black hole or a strongly distorted patch in the zenith or nadir, especially if the camera rig has no cameras pointing up/down. This issue can be mitigated by switching the *Fill Holes* parameter to *Extrude*, and possibly also reducing the crop circle size. A patch with a logo can also be placed over a nadir, adding it with a mask as one extra 'camera', switching this 'camera' to *Overlay* mode and rolling it down by setting its *Pitch* to -90. You can also switch the mapping of this 'camera' to *Planar Rectilinear*, since you probably want to make it look flat in the VR view.
- **Vertical Balance:** When you shoot in a common room, the floor and the ceiling are at approximately the same distance, and the scene is “vertically balanced”. However, in most scenes, the floor is much nearer than the ceiling (or, directly, the sky). With such an “vertically unbalanced scene”, most, if not all, calibration methods (APG, PTGui, cameras built-in auto-calibration, Mistika's “Improve”) will try to converge on both ceiling and floor at the same time. To achieve that, the calibration will result in all cameras pointing at the horizon with a slight pitch downwards, with the horizon resulting 1 or 2 degrees up from where it should be. You can compensate this typical calibration imprecision by using the “vertical offset” parameter, to place the horizon where it should be, at the center grid line of the 360x180 degrees LatLong stitched result.

SGO


# Performance considerations

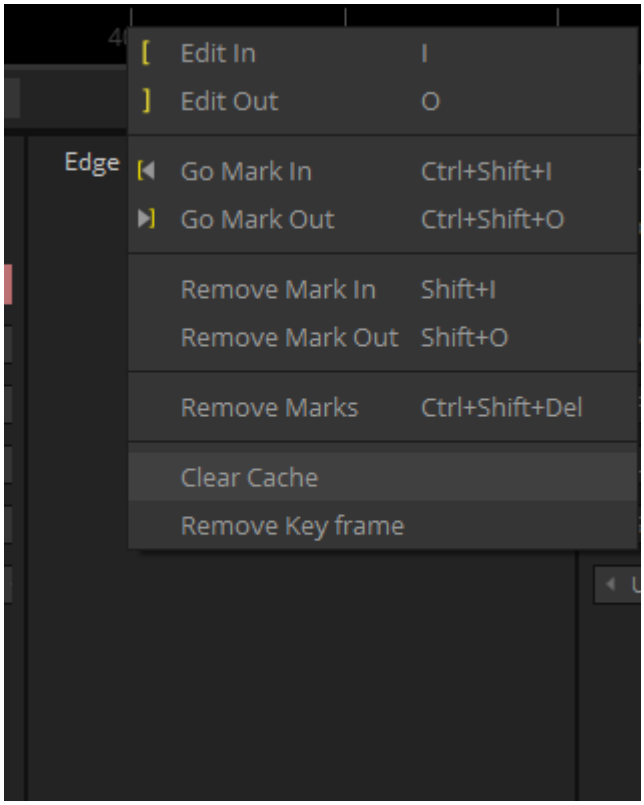
# 13 Performance considerations

### RAM Cache

The small green dots that turn into a line in the time bar represent frames already loaded in RAM. The first playback may be slow when using complicated formats, but after the images are loaded this first time they are kept in the RAM cache for subsequent playbacks, permitting realtime playbacks for very complex formats. Once the available RAM is full, older frames are automatically removed from the cache to make space for new frames.


### Clear cache

The contextual menu of the time bar has a *Clear Cache* option, allowing a *flush cache* action. This is typically used for refreshing when some cached images have been changed by other collaborating applications.



### Cache indicator

The small green dots in the time bar represent frames already loaded in RAM. The first playback may be slow when using complicated formats, but after the images are loaded this first time they are kept in the RAM cache for subsequent playbacks, permitting realtime playbacks for very complex formats.

Once the available RAM is full, older frames are automatically removed from the cache to make space for new frames.

### Recommended hardware configurations

Mistika VR is designed to run on even the smallest computers, but it will also use all the resources available to it. The following are the important considerations:

- **RAM**: if possible, install enough RAM as to permit caching the longest shots to be used.
- **GPU**: this is important for high-resolution viewing, and also for the color and framing processing of Mistika VR Controls, and for decoding some codecs (such as R3D).

Uncompressed formats (such as DPX, EXR uncompressed, or Tiff16): use storage devices which are as fast as possible (such as SSD NVMe or disk arrays), and try to use local copies rather than cloud storage. For complex cases, please note that SGO provides storage solutions that permit to playback up to 8K uncompressed in realtime.

Compressed formats (such as EXR compressed, H264/H265, R3D, or Prores): these formats depend mainly on the system's CPUs, although in some cases the GPU is also used. We recommend using the OS Task Manager or similar tools to discover where hardware bottlenecks exist for your particular case.

### Minimum Configuration Required for Mistika VR:

This is the **absolute minimum**:

- Number of processors: 1

- Number of cores: 2
- RAM: 8 GB
- Display resolution: 1920x1080
- Graphics card model: NVIDIA is recommended as it is the certified manufacturer for Mistika VR. However, according to our client reports, modern AMD boards and Intel Iris are also working without presenting problems at the time of writing.

As a minimum requirement, Mistika VR should work with any graphics chipsets if they have the following capabilities:

- OpenGL version: 2.0
- Shader (GPU): shader4 (GL_EXT_gpu_shader4)

### Recommended Configuration for Mistika VR

Below, we specify some recommendations for selecting optimal hardware:

- Most of Mistika VR's processing is carried out by the GPU: Geometry adjustments, Color adjustments, Optical Flow (typically a bottleneck), etc., so this is the most important component. A modern NVidia board with high specs is recommended.
- The only processes that are carried out by CPUs are the decoding of camera codecs and encoding of the render codec, with the following exceptions:

  NVidia H264/H265 codecs are rendered by GPU (the GPU must be NVidia, and the computer must be Windows, as Apple does not currently support it). The NVidia encoder uses dedicated hardware for this (ie. it is not used for any other task). This means that using it will not reduce the speed of other processes. If you want to take advantage of this capability, we recommend a Pascal generation or later NVidia model, as only these can carry out hardware encoding at 8K resolution. (For more details see NVIDIA VIDEO CODEC SDK).
- This is for encoding the render format: reading H264/H265 camera rushes may still require a lot of CPU power, depending on the number of cameras and their resolution.
- Uncompressed formats do not require either significant GPU or CPU, but they do need a great deal of disk speed!

Therefore, each case depends on its particular circumstances. Workflows with lots of camera rushes based on compressed codecs and heavy CPU encoding to Prores may take advantage of having many CPU cores at high clock speeds. On the other hand, stitching smaller camera rigs to NVidia H264/H265 deliveries will not take advantage of very high-end CPUs. Mistika VR uses only the GPU on which it is launched. Extra speed is cannot be expected by having two GPUs in use.

The following is another recommendation: in general, the highest end models of CPUs and GPUs offer only a little more speed than marginally inferior models, but often cost much more. At the highest end, a typical figure might be to pay up to 100% more for just 25% more speed, compared with cost-efficient hardware in the sweet spot. For this reason, in general, it may be better to use two computers with good, but not necessarily top of the range, specs rather than one only workstation with top specs., and then use distributed render tools such as Smedge for render management.

- RAM: 8GB is the minimum for simple cases (ie. few source cameras and rendering to 4K). 64GB is a good reference for most other formats, but it depends on the number of cameras and render resolution. More RAM may be required for very complex cases.
- Display resolution: 1920x1200. We do not recommend 4K GUI displays (or at least, only when having large displays of 37" or more), because the GUI elements are not yet scalable, and will therefore look too small (this capacity is planned for future versions, but not available a the time of writing).
- GPU dedicated video memory: 8GB or more is recommended when there are many cameras and when rendering to high resolutions, such as 8K+). NVidia boards are recommended, especially the RTX family (the faster, the better).
- OpenGL version: 2.0
- Shader (GPU): shader4 (GL_EXT_gpu_shader4)
- Storage: ideally, as fast as is needed to read all the camera shots simultaneously. It is also important when rendering to uncompressed formats. Please note that rendering involves both reading and writing at the same time, so if storage is not fast enough, put the original files on one drive and render to a different one, as this might improve the processing.

### Supported OS distributions (Last update: Mistika VR 8.8.8)

- **Mac**: Sierra, High Sierra, and Mojave (Operating system versions prior to macOS Sierra are known to crash the application)
- **Windows**: Windows 10 (64bit)

**Notes:**

- At the time of writing, NVidia hardware encoding (typically five times faster for encoding H264 and H265/HEVC) is not supported by Apple, and is only available on Windows (this may change over the time).
- Mistika VR is not currently officially supported on "Virtual Machines" although they should work well, as long as the license server is in a physical machine. All Mistika VR licenses are floating licenses, so a virtual machine can be configured to obtain the license from a physical system.

SGO


# Mistika VR advanced workflows

# 14 Mistika VR advanced workflows

## 14.1 How to insert an overlay clip in a 360 scene: adding logos or CG

With this procedure, you can insert an image into the scene, optionally with an alpha channel.

1. Add the image or clip you want to overlay as an additional camera, and set its *Compose* mode to *Overlay*.
2. Set its mapping to rectilinear if you want to make it flat.
3. Use its *Yaw*/*Pitch*/*Roll* parameters to locate it, and change its *Focal Length* to scale it.
4. For additional control, use its *Lens Center* to shift it left/right or up/down along its plane.

**Note**: In the case of Stereo 3D, first adjust one of the eyes, for example, the left. When this is done, add the image for the right eye, and *Copy*/*Paste* all the parameters that have been modified so it appears in the same place. Finally, set the camera's *Stereo eye* parameter to left and right, respectively.

## 14.2 Fixing vertical parallax for VR 180 Stereo3D

The *Alignment Mode* tool is now represented by an icon with cyan/magenta arrows. This tool was introduced in Mistika VR v8.8.7 and its main use is in fixing vertical parallax issues found in Stereo 3D workflows. It works for many types of rig, although it is only simple to use for VR180 parallel pair rigs.

With this new tool, the user can selectively drag three different vertical areas of each camera: *Center*, *Left*, and *Right*. While doing this, Mistika VR modifies a combination of camera parameters: *Yaw*, *Pitch*, *Roll*, *OffsetX* and *OffsetY* to affect the selected area, while affecting the other two areas as little as possible.

- First, always center the crop circles for both cameras as well as possible, as these are the main sources of misalignment and are therefore always the basis for good results.
- Enter *B&W anaglyph* mode to see any vertical misalignment.
- We recommend you choose a frame that has no object near the exact center.

- Enter/Exit *Alignment Mode*, by toggling the *AM* button (the icon with cyan/magenta arrows next to the overlay modes) in the toolbar.
- Please note that this mode only works correctly if the output camera is unmodified. If the horizon has been aligned, use *Bake in output camera* first.
- Select the camera to be moved by Shift+Clicking on its center. (If the camera has too much overlap with another, then remember that you can also select the cameras in the *Camera list* panel on the left).
- Move the camera center into alignment: in *B&W anaglyph*, the image should become black and white, without any color fringing. For the center area only you will be able to align in both axis (*Yaw* and *Pitch*).
- Follow the line of the horizon to one side (either left or right, the order is inconsequential) and drag the mouse up/down until it aligns vertically. The best methodology is to use either horizontal objects, or clearly defined objects which can easily be seen to be at the same height. Try to align using objects that are close to the horizon, because higher/lower objects may be misaligned due to the 'interaxial distance' phenomena. Some horizontal parallax will usually remain as colored fringing, because the sides are not necessarily at the same distance as the center.
- Do the same for the other side.
- It may be a good idea to re-check and realign the center and both sides again: while this tool tries to move the zones as separately as possible, it does not yet do this perfectly. An example of use can be found in the following video at 01:28 - What's New in Mistika VR 8.8.7?

## 14.3 Mass import of multiple shots and multi-segment shots

**Mass import**: drag & drop the main folder containing all the media files and as many subfolders as are needed, and the material will be automatically organized into clips. Also, an import dialog will for optionally specifying the camera preset for the camera rig in use; this will be applied automatically to each shot.

**Multipart import**: during mass import, incoming clips that have been split into multiple parts (typically due to the file size limits of SSD cards) will be automatically placed as consecutive stacks (this works for most filenaming structures ). Then:

1. When importing the folder, Mistika VR will ask for the *Number of cameras*. Set this value to the number of cameras in the camera rig in use so that Mistika

VR knows that each group of this number of files belong to a same shot, and so it will be able to distinguish between different segments.

2. After importing multi-segment shots, you can reassemble a shot by clicking on its first segment, then right-clicking on its last segment and use the *Merge from selected* option, which will combine the multiple segments into a single clip.

**Still-image panorama**: when importing an enumerated sequence, the user can choose if the images are a time sequence or if each file is a separate camera view. The following is a tutorial regarding the use of the above tools: Mistika VR 8.8.6: Workflow Improvements.

## 14.4 Integration with external render farms: Add to render queue

The *Add to render queue* button in the render panel permits sending a batch of Mistika render jobs (.rnd files) to external render farms, or transferring those files to user defined scripts. Please note that this tool requires some preparation and it is not really necessary for simple cases. If what is required is only to render a batch of shots, simply drag & drop the next camera files to the Timeline to create new shots. Then activate the All Shots option in the render panel and it will render all of them.

If more powerful render management is required, however, then the *Add to render queue* button is the appropriate tool, in combination with an external render manager.

***Add to render queue* Main Purposes:**

- To integrate with external batch manager tools and external render managers. These are specialized applications that permit organizing render jobs in different ways. For example, render jobs can be added to render queues for posterior processing (such as overnight), change their priorities on the fly, stop/resume jobs as needed, etc.
- To use an external render manager to render a single enumerated sequence (eg. .dpx, .tiff, .png) in parallel on multiple computers (with each node rendering different frames of the same clip)
- To use an external render manager to render a different movie file (eg. .mov, .avi) on each render node. Movie files can not be rendered by parts

(only enumerated sequences can) but can still be distributed by ordering one movie file to each render node.

- To execute user scripts to automate production workflows (for example, to deliver VR stitches that have already been finished to a Mistika Ultima/Boutique system).

**Configuration on the Mistika VR side:**

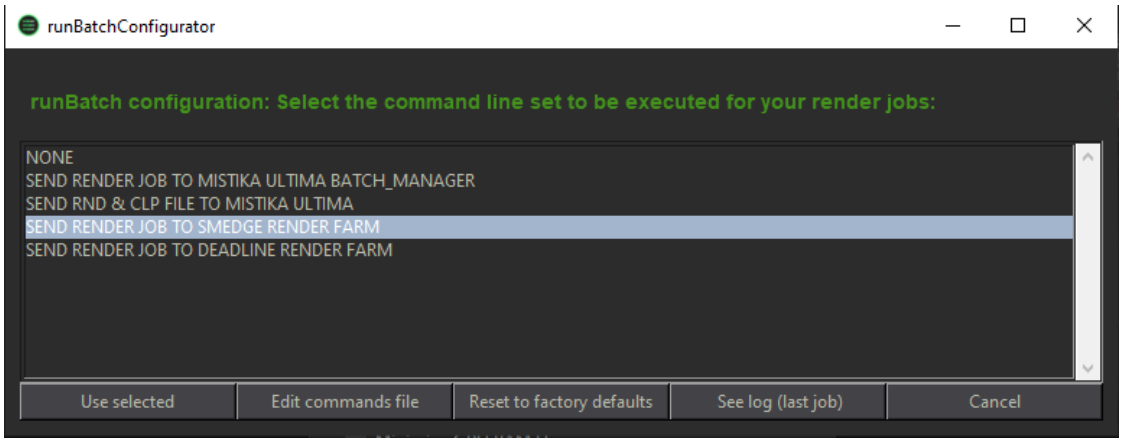
The Add to render queue button works both in Current shot and All shots modes. In the second case, a whole batch of Mistika render jobs (.rnd files) is created first, and then a set of command lines are executed for each of the .rnd files in that batch. The command lines are stored in a configuration file, runBatch.cfg, which can be configured in the File>Options menu. This menu provides a graphics interface to integrate with the most common render managers, such as Mistika Ultima BatchManager, Uberware Smedge, and ThinkBox Deadline.

For more advanced use or to integrate with other applications, the user can put all the command lines to be executed in the runBatch.cfg, transferring the path of each .rnd file and also any parameters requiring variable substitution (the full list of available variables is given in the configuration file as a reference) to them.

**Supported render managers:**

A render manager is an application that provides batch processing for render jobs, and also the capability to distribute render jobs across multiple computers, monitoring their progress and managing all aspects of the render nodes. Most of these applications are agnostic about the actual render software, which needs to be installed separately. In this context, the actual renderer (Mistika VR, in this case) can be connected to the render manager by providing a description of its command line interface (CLI), which requires a different configuration process for each of those applications (as explained below). From that point, using the Add to render queue button, the user does not need to type a command line each time. Once the commands are defined in the configuration file, the Add to render queue button will execute them for each render job.

The following are the render managers available in the *File>Options>BatchManager* menu by default:




### 14.4.1 Mistika BatchManager & Mistika Ultima

This is the usual render manager available in all Mistika Ultima environments (these render farms are Linux only, but .rnd jobs can still be submitted to them from any other platform).

Once selected, the Add to render queue button will copy the .rnd files to a Mistika Ultima folder (either to render them with Mistika BatchManager or to open them in Mistika Ultima). The configuration file also offer parameters for path translation, so it can replace local paths for the corresponding paths in Mistika Ultima systems (the complete instructions are found in the runBatch.cfg configuration file)

*Note: Another use of this mode (another example in the configuration file) is for sending .rnd jobs to a folder which is accessible to a Mistika Ultima/Boutique system in order to be opened on it, rather than rendering them directly, while producing path translation as necessary. This workflow is equivalent to opening a Mistika VR sequence in a Mistika Ultima/Boutique system, but without the need to relink the media files, as the path translation is made on the fly during the copy process.*

### 14.4.2 Uberware Smedge

This is a third party multi-platform render manager, developed by Uberware. It works on Windows, Mac, and Linux. It is probably the easiest render manager to setup, as it provides a lot of automated configuration. For this reason we have also tried to simplify the Mistika VR side of the integration as much as possible. It should work as follows:

1. First, make sure that all the computers are sharing a common storage volume for the source files and rendered files. Check that Mistika VR can read the media files from those volumes, and also render on them, otherwise the following steps will not work properly (except if you are installing Smedge on the local computer only, as a simple Batch processor).

2. In all the render nodes, install both Mistika VR and Smedge (Smedge can be found at www.uberware.net). Make sure that all nodes have the corresponding necessary licenses for both products, and test that they can render using Mistika VR as usual, without activating the *Add to render queue* function yet (this is because, obviously, it does not make sense to try the integration functions with Smedge until Mistika VR is rendering correctly by itself).

3. Copy the attached Smedge modules available in this document: Integration with external render farms (Add to render queue) (MistikaVR.psx and MistikaUltimaAndBoutique.psx) to the "Modules" folder of the Smedge installation. The .psx module file is the same for Windows, Mac, and Linux; only the Modules folder location will change. Default paths are:

- **Windows**: C\Program FIles\Smedge\Modules
- **Mac**: /Applications/Smedge.app/Contents/MacOS/Modules/

The MistikaVR.psx that comes with Smedge 2018 is only valid for Mistika VR versions prior to v8.8.6. For v8.8.6 and later, please download the latest .psx module. Smedge provides its own system for path translation, in case that the file paths are different on each of your systems. For more details please search for "Path Translation" on the Smedge documentation.

Then start Smedge (or restart it if it was active, as this is necessary to refresh the modules list). From that point, Smedge should already offer "MistikaVR" and "MistikaUltimaAndBoutique" as available job types. (for more details please read the Smedge documentation about creating jobs).

At the date of this document (Mistika VR 8.8.6) the MistikaVR.psx module that comes with Smedge is obsolete, please find the latest one in the Mistika knowledge base, it is kept updated at the end of this document: . Then you need to totally exit from Smedge and open it again (or reboot the system if you are not sure). This is necessary in order to refresh the modules list). From this point on, Smedge should already offer Mistika VR as an available job type. For more details, please read the Smedge documentation about creating jobs.

4. In Mistika VR, select File>Options>BatchManager. This will offer several configuration options: choose Send .rnd jobs to Smedge. From this point, every time you click the Add to render queue button it will submit the render jobs to Smedge (one per shot), rather than rendering them directly. Later, if you want to use a different render manager or add user scripts, change it in the File>Options menu.

The Smedge progress bar is based on counting the number of rendered files. Enumerated sequences will show a progressive bar, but movie files will jump from 1% to 100% in one step. This is the expected behavior, as a movie file is just one file. However, you can still see the frame number being rendered in the metadata columns at the right of the progress bar.

### 14.4.3 Thinkbox Deadline

The instructions for integration with Thinkbox Deadline are in the Deadline 10.0 Documentation: Deadline 10.0.23.4 documentation

Please note that Mistika's executables path has changed since the original document. You will need to adjust to your specific installation paths.

### 14.4.4 Compatibility with other render managers and workflow automation tools

In principle, Add to render queue function should be compatible with any render manager that permits submitting jobs by command line. This is because this tool can carry out one or more user-defined command line for each .rnd job, also transferring the path to the .rnd file as a parameter and performing variable substitution for other common parameters in the command line (start frame, end frame, media paths, etc). However, on the other side of the connection, many render managers also require a plugin or a configuration file to describe the command line interface (CLI) of each product that they support. If your render manager does not have your Mistika product on the list, or if you are integrating with your own workflow tools, then you will need to make that missing piece, for which you can find the necessary information in the chapter Render CLI: Rendering with command line interface. 130

**Support services and trouble shooting:**

A characteristic of most render farms is that they are based on a combination of products from different manufacturers. Given this fact, in the case of problems, the first thing to do is to isolate which product is causing the problem (logically, each of the providers can only help you with the configuration of their own products). We recommend the following procedure to locate the source of the problem and, therefore, which support service needs to be contacted:

**Recommended support procedure:**

First, check that all network and shared storage volumes are working correctly on all computers before installing any render farm management software: network and permission issues are the most common source of problems in a render farm. Specifically test that you can can access all source files and overwrite render folders from all the computers without permission problems. If problems are detected in preliminary sequence of tests, contact your system administrator, and do not continue to the following points until this part functions perfectly. It is highly recommended to start with a working network and fully operational network drives before installing any render farm manager software on it, otherwise they may fail to execute any automated configuration provided by their installer tools, and when the network is fixed, you may have configuration issues that cause the need for manual tweaking for everything.

If the problem has appeared at a later point, a useful test is to configure the local system as the only render node, and then test the connectivity between Mistika VR and Smedge (or whichever render manager is being used) locally on that single node. If the integration works well on each computer but not when added them together, then the cause of the problem is probably a network issue. In particular, check that all your computers are discoverable in the network, as the render managers will probably use that function (in the case of Windows, make sure Network Settings>SharingOptions>Turn on Network Discovery is activated). Also make sure that firewalls and antivirus software are not blocking any necessary connection (if possible, deactivate them, and reactivate them when it works).

Once the potential problems above have been dismissed, check that Mistika VR can render normally on each render node using the shared network drives (without using Add to render queue or any render manager yet). Once that Mistika VR standalone render is working well on all computers using shared volumes, then the only process left is the render manager configuration. That should be the last piece

to be installed and configured, and their documentation and support services will be the ones to be used for this purpose.

**Note for users with installation paths different than default:**

The .psx module file will tell Smedge how to build the path to find the VR renderer. For example, if your VR executable is in this folder: *C:\Program Files\SGO Apps\Mistika VR\bin\vr.exe* it will use a .psx file with this section:

```
[ FindLatestExecutable ]
WindowsRoots = C:\Program Files\SGO Apps
WindowsBases = Mistika VR
WindowsExe = bin\vr.exe
```

## 14.5 Collaboration between Mistika VR and Mistika Ultima/Boutique

Mistika VR projects can be loaded on Mistika Ultima (v8.7 and later) or Boutique for advanced post-production tasks that are not possible in Mistika VR, including the following, among others:

- Online editing
- Stereo 3D advanced adjustments & Depth Grading
- VR multilayer compositing and CG integration
- Animated VR shapes
- Motion Tracking
- Color grading in VR space
- Client-attended sessions, optionally with live streaming to VR helmets
- Conform tools for EDL, AAF, XML
- Paint, Warp, Morph, Titling
- Finishing and VFX workflows
- Faster than realtime rendering with Mistika Totem multiGPU technology

For integrated workflows, the most important consideration is to use versions of both products from similar dates. The version build date is detailed in the distributions of both products, and also in the *About* menu (Mistika VR) and *Status Bar* (in **Mistika** Ultima/Boutique).

**Sending VR projects to Mistika Boutique**

1. Save your environment on a path with permissions for Mistika Boutique. It is recommend to make a *Save sequence as* back up.
2. Open Mistika Boutique. Now you can drag and drop your environment or just select the file and double click it to open it. Now Mistika Boutique will open a timeline with all the clips and all the stitching information.

All the parameters are translated to Mistika Boutique, and can be manipulated as well, there is no need to go back to Misitka VR. The MIstika Boutique timeline have much more flexibility for moving the clips, trim them, group them...etc.

**Sending VR projects to Mistika Ultima**

1. Transfer the Mistika VR project to a Mistika Ultima system, and load the .vrenv file.
2. If necessary, use the *Relink* tool to find the media files in their new locations (see the following point for a more automated workflow and automated path translations).

Alternatively, in Mistika VR you can use *VR>Render>AddToQueue* to send .rnd files to Mistika Ultima/Boutique. This can be used to render them with BatchManager, or to make the .rnd files availalble to be opened in Mistika Ultima. The *AddToQueue* integration is explained in the previous chapter. [121]

**Path translation:**

All Mistika applications from v8.8.6 and later support path translation between different systems, which works as follows: when a media file is not found (for example, if it had been located in a different path in the original system) then the Mistika application will ask the user for the new location. Once the user has answered this for the first clip, Mistika will save this information in a configuration file and try to use it as a template for cases arising later.

So the easiest way to configure a multi-platform environment using different paths on each system is to transfer an example project to the others and answer the

prompts when loading it. Mistika will then learn from these examples for solving future cases and no more prompts/questions should appear.

If that is not enough, the *FileBrowser>Relink* tool permits establishing the new locations manually.

**Integration trough the Mistika BatchManager:**

In Mistika VR, the render panel provides an Add To Render Queue button[121]. Among other options, this tool permits sending .rnd render jobs to a Mistika BatchManager, which is the render manager provided with Mistika Ultima. You can use this tool in two ways:

1. to render .rnd jobs with Mistika Ultima's renderfarm (or any other supported render farms).
2. when a stitching job is finished, send it in an .rnd file to Mistika Ultima for additional conform and post-processing tasks. The workflow would be identical to the previous case, but by choosing an inactive queue (no render done).

**Scripting:**

The .vrenv file contains all the work done in Mistika VR. Because .vrenv are text files, it is easy to create custom scripts in order to automatize workflows. The syntax is relatively obvious, although more information can be found in the Chapter Mistika scripts syntax.[132]

**Other resources:**

- **Mistika Ultima user**: at the time of writing, Mistika Ultima v8.7 (the first version with VR support) is still Beta. Mistika Ultima betatesters should contact SGO support to ensure that they use compatible versions if they plan to use combined workflows.
- **Mistika VR user**: to receive Mistika Ultima services without purchasing them, ask in the SGO forums for Mistika Ultima facilities available in your region. Many Mistika Ultima facilities around the world already offer advanced post-production services for Mistika VR files.

## 14.6 Render CLI: Rendering with command line interface

This section is only about the basic command line renderer. Integration with external render farms is described in the chapter Integration with external render.[121]

Whenever the render function is used in Mistika VR, a render job is saved to a text file (with an .rnd extension), in the *DATA\RENDER\RenderName* folder of your project. These render jobs can also be rendered with the "...*vr*" *-r* command line.

There are two ways to produce .rnd files:

1. The most obvious method is to click *Cancel* as soon as the progress bar has appeared. The render will be stopped, but the .rnd files will be already created for later use.
2. A more advanced method is to use the Add To Render Queue[121] button, which also permits sending the .rnd files to external render farms or user scripts.

**Command line syntax:**

- Windows:

**"C:\>Program Files\SGO Apps\Mistika VR\bin\vr" -r Path_To_Rnd_File [ -s start_frame ] [ -e end_frame ]**

- Mac:

**"/Applications/SGO Apps/MistikaVR.app/bin/vr" -r Path_To_Rnd_File [ -s start_frame ] [ -e end_frame ]**

- Linux:

**"$HOME/SGO Apps/Mistika VR/bin/vr" -r Path_To_Rnd_File [ -s start_frame ] [ -e end_frame ]**

The following is a real render example on Windows, with all the necessary paths in place:

**"C:\>Program Files\SGO Apps\Mistika VR\bin\vr" -r C:\MyProjects\MyProject\DATA\RENDER\testrender\testrender_0000.rnd -s 200 -e 300**

Parameters '-*s*' and '-*e*' define the start and end frames and are optional (the whole clip is rendered if not included) but note that they are only available when rendering to enumerated sequence formats (eg. .dpx, .tiff, .jpg). This is because movie formats (eg. .mov, .avi.) cannot be rendered by parts, but only as a whole. Also, those parameters must belong to the frame range that is covered by the rendered segment, which is normally defined by the *In* and *Out* marks set in the timeline. For example, if *In* and *Out* marks are set from 100 to 200, you cannot indicate a start frame before (below) 100 or after (above) 200 in the command line. In another example, many command lines can be put in a .windows .bat script for batch processing, (or an OSX shell script for Mac), and then the script can be executed overnight to render all jobs one after the other. For more advanced use, see the document on Integration with external render farms: Add to render queue.[121]

## 14.7 Working with heterogenous rigs combining different camera models

Heterogeneous rigs can be complicated, but they are still possible. A good approach is to combine two different presets:

**Example**: Using Z Cam for main cameras + GoPro for nadir/sky:

- First, stitch the Z Cam footage using the Z Cam preset.
- Then, poach camera settings for the GoPro from a GoPro based preset (eg. the Omni). Apply this preset (it will become completely broken), locate the *Camera Defaults* tab and take note of the relevant values: *Focal length* (16.449); *Lens distortion* (a,b,c = 0.0, 0.041, 0.0); and *Circle Mask* (100 - stock GoPros are full frame).
- Next, *Undo* to return to the Z Cam stitch, and add the GoPro camera to the stack. Select it, and in the *Input Camera* tab, input the above data only for this

camera. (Do not input this data in the *Camera Defaults* tab, as this will affect all cameras, which is not desired.)

- Set *Pitch* to 90 (additional GoPro pointing upwards), and increase *Roll* until you find an approximate match with the rest of the cameras.
- When the cameras are a match to the naked eye, use *Improve Angles* to refine the added camera heading.

## 14.8 Mistika scripts syntax

There is no formal documentation about Mistika scripts syntax. However, as everything is saved in text files, in general, it is easy to understand syntax by saving simple example timelines and reading the content. Although they are always text files, an important detail to remember is that there are two possible formats: *compressed* and *uncompressed*. By default, Mistika produces compressed files in order to obtain better performance when reading and saving files.

**Compressed and uncompressed formats**

This point only applies to Mistika Ultima syntax. Unlike Mistika Ultima, Mistika VR always uses the uncompressed format (.vrenv scripts are always small, so they are not compressed). In compressed (or, more accurately, 'compact') format, each effect only appears completely the first time. From that point onwards, all animation curves that have default values will simply appear as *c()*. This normally reduces the size of the file a great deal, although it complicates reading them from user scripts, because the number of *c()* fields must be counted, rather than searching for their names. To switch to uncompressed, set this variable environment variable in your operating system:

**COMPACT_CURVE_SAVE 0**

In the case of Linux, there are two procedures:

1. SUSE11 systems (.cshrc file): setenv COMPACT_CURVE_SAVE 0
2. CENTOS systems ( .bashrc file): export COMPACT_CURVE_SAVE=0

**Effect numbers, effect names, and effect definitions**

The prototype definition for each effect is stored in *MISTIKA-ENV/etc/curves* (where MISTIKA-ENV is the path to the software location, which will depend on the

product and OS). Use these files to avoid having to save examples. Also, use the filename to find which effect number corresponds to it. It is also useful is to check if there are new parameters in a new version, in order to automate changes to the scripts. In these files, the default parameter vales are the ones inside the *d()* field, which is the only value in these .proto files. Once in a .env file, each *d()* field is then followed by the user keyframes

**Plugins and Inline data**

- Shapes are embedded in the .env files converted to text.
- For performance reasons, more complex binary data (typically from external plugins) is not saved in the .env files. Instead, it is saved in the *PRIVATE* folder of the project, with the name of the environment as part of the filename. In terms of Mistika effects, only VectorPaint, Title, and Morph use this method to store vectorized information. Other plugins may have their own files located in there.
- Use *backslash* as an escape character for reserved characters in media paths and comments
- The characters "(" & ")" are reserved for syntax, as are "{" & "}" which can create problems with filenames. However they can be supported by using the *backslash* "\" character.

**Example:** MyCLip\{1\}.mxf

The *backslash* "\" character is safe to use for this because all paths are normalised to Unix syntax, even for the case of Windows filesystems (where they will be converted to "/"). So Mistika will never insert a "\" character in a path. Most syntax is insignificant, but some aspects are not obvious. The following is a compilation of FAQ answers to support cases related to the syntax.

**Question**: When populating timelines with scripts, which parts of the scripts can potentially be omitted?

**Answer**: The whole *AudioRoute(...)* section can be omitted, including the clip's *AudioRoute(...) local statements*. Mistika will set it to default, which is okay.

**Question**: What are the *L()* and *H()* fields?

**Answer**: The *L()* and *H()* statements refer to *Hi res* and *Low res* (proxy) media. Currently, it is not normal to have any proxy media in modern systems. Mistika can

create proxies on the fly in realtime (and then process the effects in proxy resolution). Then, for example, simply leave the proxy line with "?" values:

L( d(Xfs.dev) p(?) n(?) f(?) )

**Question**: Can can we include our own metadata somewhere?

**Answer**: The *c(..)* field in each clip allows *Comments*, which is the easiest way to do this. Mistika will interpret it simply as that, as a comment. You can create examples with *Edit>Attributes* to see how they work.

But there is a case in which Mistika also uses them for another purpose: in the case of R3D clips, Mistika will create a snapshot of the R3D settings at the moment of R3D import. However, the actual R3D settings for processing are always read from the R3D media itself. If you want to keep this metadata as a reference, create an example R3DParams effect. Round brackets - "(" & ")" - are obviously reserved characters for the script syntax. If you want to use them inside comments or elsewhere, it is still possible, but they need to be escaped as "\(" and "\)".

**Question**: How are R3D multisegment movies represented?

**Answer**: In the case of multisegment R3D movies, you need to indicate the first one (the 001.R3D) . The codec itself will then follow the other segments. For example:

H( d(Movie.dev)
p(/TRANSPORT/CUSTOMERS/SUNSET/C005_L008_061459.RDM/C005_L008_061459.RDC)
n(C005_L008_061459_001.R3D) f(?)
)

**Question**: What is the meaning of the *I()* section?

**Answer**: *I()* stands imply for *Image()* as opposed to *Sound()*. Audio clips will use *S()*.

**Question**: What are all the coordinate sections or sections containing numbers?

**Answer**: Inside a clip, the statements mean:

X() - the frame of the timeline; grows rightwards
Y() - the layer of the timeline; grows upwards
The timeline coordinates are signed 32 bits numbers, and origin can be chosen arbitrarily, although for it is good practice to start somewhere near (0,0).

For example, choose the *Y(0)* layer for the right eye and *Y(1)* for the left eye on top of it. If you needed to add effect nodes on top of each clip, leave some space, specifying, for example, *Y(0)* and *Y(5)*.

For *X()*, the first clips start at *X(0)*, the second at *X(1234)* if the length of the first shot is 1234 frames, etc.

*W()* stands for the *Width* of the clip as drawn in the Timeline. It equals the number of frames of the media, so if the head or tail needs to be trimmed, both must be subtracted from this width.

*f(0) l(1216)* - first and last frame of the trimmed media; first: inclusive; last: exclusive.

*s(0) e(1214)* - start and end index of the original, untrimmed, media, both INCLUSIVE in this case.

**Some examples**:
The media has 1000 frames. The untrimmed clip would state (in the correct places of the structure):

W(1000)
f(0) l(1000)
s(0) e(999)

The same clip with 5 frames trimmed from its head (beginning):

W(995)
f(5) l(1000)
s(0) e(999)

The same clip with 5 frames trimmed from its head (beginning) and 10 frames trimmed from its tail:

W(985)
f(5) l(990)
s(0) e(999)

*t()* is the tapename. For example, t(C005_L008_061459).

*T(10:29:41:05)* is the Timecode of the first frame of the media. Trimming does not change this number.

Timecodes can vary depending on media file type, so it is recommended to obtain an actual reference Timeline with clips referencing real media. Here is an example of an actual *T()* statement for an R3D timecode:

T(1950570@47.952049|1950570@47.952049|172801@47.952049)

Firstly, the "|" character separates multiple timecodes available for the media (there are three of them for R3D, the first two normally identical) so the string describes three different timecodes, in this case:

1950570@47.952049
1950570@47.952049
172801@47.952049

The number before the "@" character is the timecode converted to a frame index of a hypothetical sequence starting at 00:00:00:00. All frames are counted, so 00:00:00.10 would be number 20, not 10, due to the ./: TC pairing. Up to three different TCs will be displyed by Mistika's interface, but you can specify fewer, although at least one must be present. The number after "@" is the framerate of the initial timecode, as it could be different from the working framerate of Mistika at that moment.

SGO

# Mistika VR F.A.Q. (Frequently asked questions)

# 15 Mistika VR F.A.Q. (Frequently asked questions)

This part of the document covers questions that do not have their own dedicated section, so, for example, questions about licenses are not included here, but rather in the license section. The following ar a sample of recent questions from new users:

**Q: How do you select the project resolution? / How do you create a new render resolution?**

**A**: See this document.

**Q: How do you reduce text size in parameter boxes? It does not fit the field size, and the values appear cropped.**

**A**: Mistika VR uses the system font. In Windows 10, this is configured in the Display Settings. Values over 150% are too big for Mistika VR's interface

**Q: How do you sync clips by using their audio tracks?**

**A**: The *Align by audio* tool has these parameters for adjusting (in case it does not work well at the first attempt):

- *Search length* is the length of the sound sample to compare, centered at the current frame position.
- *Maximum offset* is how much the cameras may be out of sync, in order to avoid false matches at unlikely large offsets. Turn on the cameras one by one, think how many seconds they may be out of sync, and input that number as the maximum. Set the current frame at a zone where there is some identifiable noise (eg. clap, people talking), and run the match.
- *Sample size* should be left on its default setting. The audio needs to be split parts - 'windows' that can be matched - and this is the size of these parts: smaller means more precision on transients, but lower frequencies may get ignored with very small windows. Maintain a range of 2 - 4, which works best. This field should probably be hidden, as it is too technical, and tweaking usually does not produce significant changes, even or make it worse, in very rare cases.

**Q: What does the "Cannot open video codec" error when rendering mean?**

**A**: It means that Mistika VR sent the rendered images to a third party encoder and it returned an error. The exact cause is not defined, but it normally means one of the following:

- An impossible setting, like bitrate too high for the selected codec (a typical absolute maximum  is 135000, except for lossless codecs), or resolution too high for the selected codec (4k for H264, 8k for H265, also depending on the GPU generation). In the case of H264/H265 (HEVC) Nvidia supported formats for each GPU model are documented here: NVIDIA VIDEO CODEC SDK.
- In the case of NVidia H264/H265 (HEVC) hardware codecs, please note that the hardware encoder is a limited resource that can not be multi-tasked (most GPUs only have one encoder). So it can also happen if another application that can use the NVidia encoder is opened at the same time.

**Q: Why does my render in the Mistika VR Evaluation version have a watermark?**

**A**: Starting from version 10.8.3, the Mistika VR Evaluation version includes a watermark on rendered content. This watermark is a feature limitation specific to the Evaluation version, allowing users to explore the software's capabilities before making a purchase. To access watermark-free renders, consider upgrading to the full version of Mistika VR.

If you have acquired and activated a paid edition of Mistika VR, but your renders still display a watermark, it indicates that you are still using the previous activation code from the evaluation period. To resolve this, please deactivate the old code and activate the new one that you received via email upon completing your purchase. For additional information, please refer to the licensing articles.

**Q: How do you animate parameters (edge points and others)?**

**A**: From v8.7.7, most parameters can be animated. To control animation, open the contextual menu for the parameter by right clicking on it. These options will appear:

- *Default Value*: resets the parameter. If it was animated, animation will be disabled.
- *Add Key Frame*: a keyframe will be added at the current frame position. Animation is enabled for the parameter if it wasn't enabled already. From this point onwards, any change to this parameter will automatically insert a new keyframe at the current time (if there wasn't a keyframe tere already).

- *Remove Key Frame*: the keyframe at the current time will be removed. If it was the last remaining keyframe, then the parameter will become non-animated, going back to the default value.
- *Remove Animation*: the animation will be disabled, with all keyframes removed. In this case, the current value of the parameter will be kept as non-animated value.
- The numerical value of the parameters is color coded:
  - o Gray numbers mean the default, unmodified value.
  - o White numbers mean the current value set by the user, but it is not animated. Please note that if it is set manually to the default value, it will remain white, because it was set by the user. To completely remove user actions, use the *Default value* command.
  - o Green numbers are keyframe values set by the user.
  - o Light Blue numbers are values interpolated between keyframes.
- In the time bar, keyframes will be shown for the selected parameters as green marks, while animated segments will be drawn as light blue segments, thus corresponding to the same color hints.

**Q: When I render a movie in Mistika VR and watch it in other application, why it is not shown as 360 video?**

**A**: You may need to insert the 360 spacial metadata into the movie file: select *Inject Spatial Media Metadata* in the render panel (available for render formats supporting it, but not all formats support it).

**Q: What is *Bake In Output Camera*?**

**A**: Whenever you move the horizon, Mistika VR does so by changing the *Yaw*, *Pitch* and *Roll* parameters on the *Output Camera*. *Bake in Output Camera* clears this parameters by adding them to the control parameters of the Input cameras instead. The rendered images will be the same, but it can be useful to do it in order to establish the new, preferred horizon settings 'by default'. For example, if you want to experiment and easily return to that setting later, or if you plan to animate it or do scripting with the metadata files.

**Q: How do you add a logo or CG overlay clip in the scene?**

**A**: See Chapter How to insert an overlay clip in a 360 scene: adding logos or CG. 119

**Q: How do you stitch a cube rig?**

**A**: Cube configurations require some puzzle solving, as the files do not come in any specific order. If you do not have a preset for your particular cube rig, start by applying the Omni or Freedom360 presets, and try to join the three views of the leg in one corner. The other three views tend to be simpler. Depending on the camera configurations, either it all fits into place, or one camera will remain flopped (180 degrees rotated). In such a case, use the other preset from those abovementioned (Omni or Freedom) next time, or simply add 180 degrees to the last camera's *Roll* value. Once you have everything adjusted, save a preset for your rig. The following is a good tutorial using a cube rig: Mistika VR Stereo Stitching. Presets and scene based optimisation.

## Frequent questions from advanced users:

**Q: How do you do a "vertical flip" to a camera:**

**A**: There is a parameter for a horizontal flip in the *Source Camera* tab. The need for a vertical flip is very rare on VR rigs so and it does not have a specific button. However, it is still possible: rotate the camera 180 degrees by adding 180 degrees to its *Roll* value, and then use the horizontal flip.

**Q: Insta360 Pro calibration data: what has happened to it in Mistika v8.8.7?**

**A**: The calibration data normally generated with Insta360 Pro Studio (the .prj file that goes as a sidecar with the media files) can now be generated automatically inside Mistika VR (similar to what was already possible for Kandao) which will simplify the workflow by removing the need to use an extra application and then using the ImportStitch to read the .prj file, which is no longer necessary. Now when using the tools for improving transformations, if *Use Insta360 Pro Callibration* is selected, Mistika VR will automatically use an integrated Insta360 Pro plugin for creating identical calibration data. However, Mistika VR's own tools are still available, as usual, and some users may still want to improve the calibration on its own. For more detailed information see the Chapter Insta360 Pro [67]. Some other cameras still come with calibration files in sidecar files: Nokia Ozo (.txt file) and Facebook surround (.json file). These can be imported with the *Stitch>Import Stitch* (or just dragging & dropping the file).

**Q: Why are there no "render only licenses" for Mistika VR?**

**A**: In our current license model, the main difference between using the evaluation version and a subscription is the capability to render. So render is the main capability that users pay for, not the GUI. This means that it does not make

sense to have a 'render only' license, because the Mistika VR GUI will work anyway. However, if you need to feed a big render farm for long periods of time, ask our sale representatives for volume discount. A typical case might be that you have a big render farm used for many products, and you only need to use some of them with Mistika VR, but you do not know which ones will be available at any time. For this case, activate all your Mistika VR activation codes on the same system (the same license server), then tell all the render nodes to use that license server, and set the policies of your render manager (Smedge, Deadline, etc) to retry periodically until the renders are complete. This way, even if all licenses are busy for a while, when one license becomes available, the next render node to try a Mistika VR job will succeed, and all render jobs will eventually be completed.

**Q: What is the difference between *Camera Default Offset-X / Offset-Y* and *Camera Input Offset-X / Offset-Y*?**

**A**: The default *Offsets*, *focal length* and *lens distortion* values are used only for cameras where the corresponding values are left unmodified. This means that once you set these values manually, or by using the *Improve offsets* tool, the *Defalut OffsetX/Y* values are ignored from that point onwards. Normally, the only common situation where default offset values are used is when importing from PTGui and *individual cameras offsets* control is not enabled there. This is normally a good idea only if you are not actually using a rig with multiple cameras, but rather rotating a single camera, panorama-still style.

**Q: How do you increase RAM cache for more interactivity and faster playbacks?**

**A**: Mistika VR will cache the most recently accessed image files in RAM memory. The cached images are represented in the timeline bar by a thin highlight bar. If you scrub trough cached images or play them back, cached versions are used, thus providing a faster response. The number of cached images depends on the amount of RAM reserved for this purpose, and it is calculated automatically to a safe setting based on your computer configuration. However, if you want to tweak it manually, you can do this in the *localPreferences.xml* file:

(Default location: \Users\<user>\SGO/AppData/VR/localPreferences.xml)

Two parameters are involved (it is risky to modify them and they can destabilize the system, so please make a backup of the file first:

- cpuFrames (maximum number of frames to cache. Leave this setting at 0 for automatic adjustment, or set another value to force it)
- cpuMemKb (RAM amount, in kilobytes)

**Q: How do you hide (or show) the color picker tool?**

**A**: This is a standard *Pick color* tool in all Mistika Technology applications to obtain the exact color values of a pixel. It does not have an icon because it is rarely used in Mistika VR, but it may be activated inadvertently. The hot key is Ctrl+Alt+Right mouse button: use the hotkey with the pointer away from the image area to hide it.

### Frequent questions about Stereo 3D:

**Q: How do you reduce parallax crossover or excessive parallax due to objects in close proximity to the camera?**

**A**: Parallax crossover is visible in *B&W anaglyph* as red/cyan lines crossing each other. When all else fails, a way to try to save the shot is to increase the feather parameter. This produces the effect of reducing all parallaxes, so you can use it to try to bring parallax to an acceptable level.

**Q: How do you use the AM tool to align vertical parallax (VR180 Stereo 3D)?**

**A**: The *Alignment Mode* tool the (*AM* icon, now represented by cyan/magenta arrows) permits fixing vertical parallax differences, and it is mainly useful for VR180 video. It is explained in the Chapter Fixing vertical parallax for VR 180 Stereo 3D [119].

SGO


# Mistika VR Shortcuts

# 16 Mistika VR Shortcuts

### File

- Load Sequence: *Ctrl* + *O*
- Save Sequence: *Ctrl* + *S*
- Save Sequence As: *Ctrl* + *Shift* + *S*
- Quit: *Ctrl* + *Q*

### Edit

- Undo: *Ctrl* + *Z*
- Redo: *Ctrl* + *Shift* + *Z*
- Cut: *Ctrl* + *X*
- Copy: *Ctrl* + *C*
- Paste: *Ctrl* + *V*
- Select All: *Ctrl* + *A*

### Tools

- Match Colors: *Ctrl* + *Shift* + *M*
- Match Color in Time: *Ctrl* + *Shift* + *N*
- Improve Offsets: *Shift* + *Z*
- Improve Angles: *Shift* + *X*
- Bake in Output Camera: *Alt* + *B*
- Add Edge Point: *Ctrl* + *Shift* + *A*
- Remove Edge Point: *Shift* + *Backspace*
- Add Edge Point Left View: *Ctrl* + *Shift* + ←
- Add Edge Point Right View: *Ctrl* + *Shift* + ←

### Time Controls

- Previous clip:  *Control* + ←
- Next clip:  *Control* + ←
- Set in :  *I*
- Set Out:  *O*

- Go to In: *Ctrl* + *Shift* + *I*
- Go to Out: *Ctrl* + *Shift* + *O*
- Clear In: *Shift* + *I*
- Clear Out: *Shift* + *O*
- Clear In&Out: *Ctrl* + *Shift* + *Delete*
- Previous Frame:  ←
- Play Backwards:  *J*
- Stop:  *K*
- Play Forwards:  *L*
- Next Frame:  →
- Go to Head: *Ctrl* + *Home*
- Go to Tail: *Ctrl* + *End*
- Next Keyframe:  *Shift* + →
- Previous Keyframe:  *Shift* + ←

## Display Controls

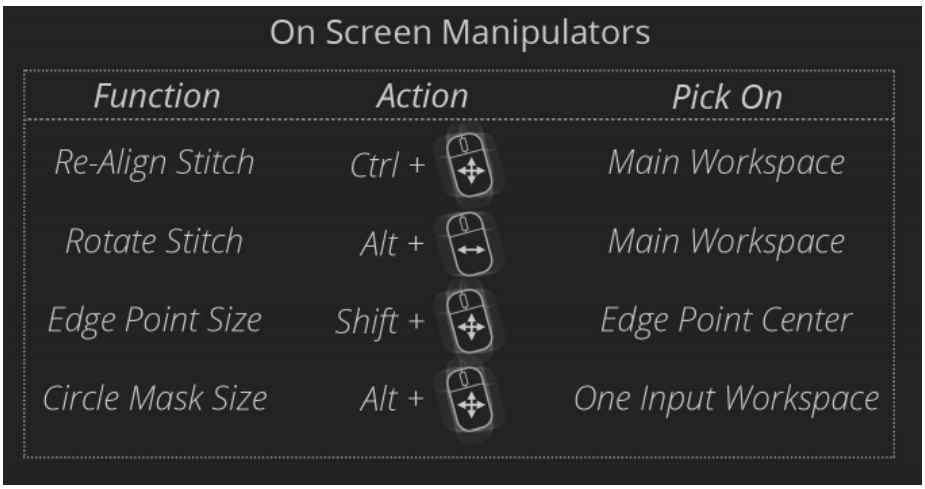
- Storyborad:  *S*
- Mosaic:  *M*
- One Input:  *C*
- FullScreen:  *Double Click*
- VR Mode:  *V*
- Camera Overlay:  *1*
- Feather Overlay:  *2*
- Grid Overlay:  *3*
- Zoom In:  ↑
- Zoom Out:  ↓
- Zoom Reset:  *Ctrl* + *Del*

- Zoom Center: *Ctrl + Ins*
- Quick VIew: *Ctrl + Alt + P*

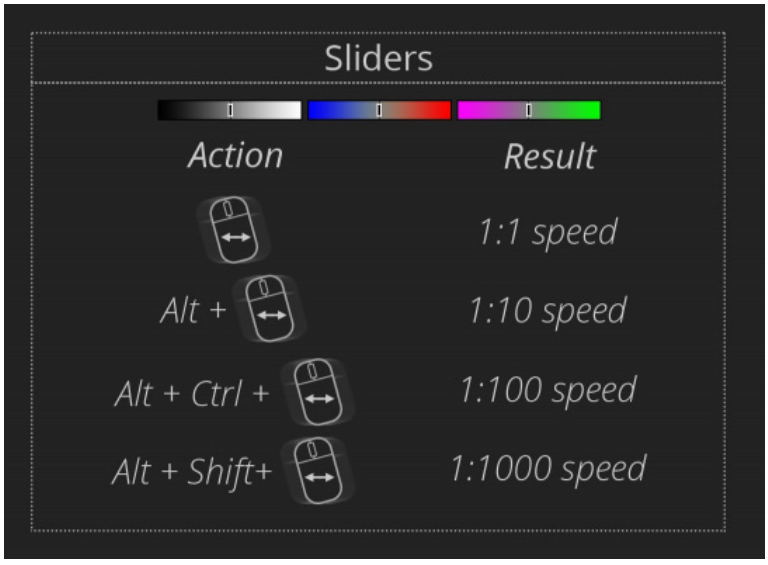
### Stereo 3D Views

- No Stereo: *N*
- Left: *E*
- Right: *R*
- Anaglyph: *A*
- B&W Anaglyph: *B*

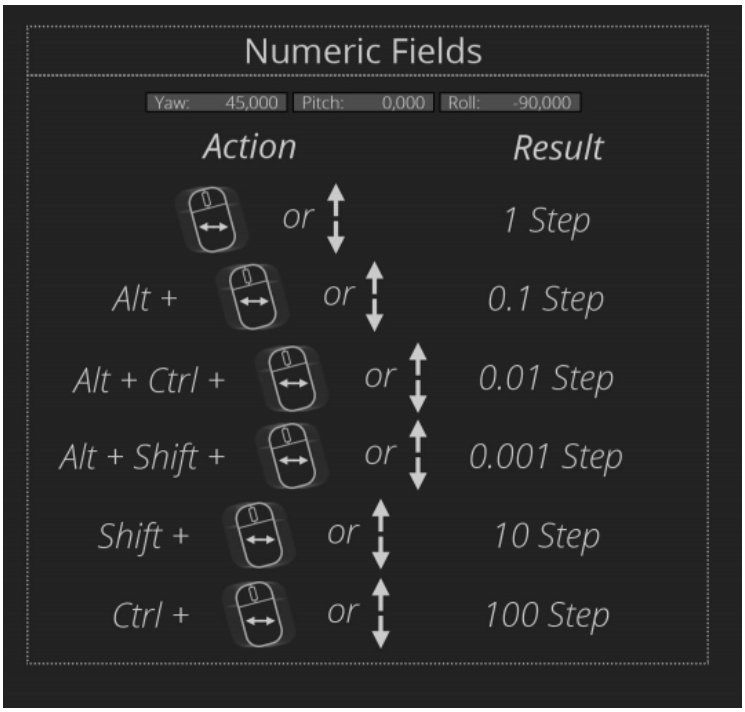
### On Screen Manipulators




### Sliders



### Numeric Fields




### Shortcut Cheatsheets

There is avaiable a a quick cheatsheet of the Mistika VR shortcuts upgraded to the version 8.10.1. Two versions included: one with white background for printing and the other with a black background for better screen visualizing. In order to access the two shortcut cheatsheets, click on the picture next page.

# Mistika VR

*Shortcut Cheatsheet*

## File Controls

| Function | Shortcut |
|---|---|
| Load Sequence | Ctrl + O |
| Save Sequence | Ctrl + S |
| Save Sequence As | Ctrl + Shift +S |
| Quit | Ctrl + Q |

## Stereo 3D Views

| Function | Shortcut |
|---|---|
| No Stereo | N |
| Left | E |
| Right | R |
| Anaglyph | A |
| B&W Anaglyph | B |

## Tools Controls

| Function | Shortcut |
|---|---|
| Match Colors | Ctrl + Shift + M |
| Match Colors in Time | Ctrl + Shift + N |
| Improve Offsets | Shift + X |
| Improve Angles | Shift + X |
| Bake in Output Camera | Alt + B |
| Add Edge Point | Ctrl + Shift + A |
| Remove Edge Point | Shift + Backspace |
| Add Edge Point Left View | Ctrl + Shift + ← |
| Add Edge Point Right View | Ctrl + Shift + → |

## Display Controls

| Function | Icon | Shortcut |
|---|---|---|
| Storyboard | | S |
| Mosaic | | M |
| One Input | | C |
| VR Mode | | V |
| Camera Overlay | | 1 |
| Feather Overlay | | 2 |
| Grid Overlay | | 3 |
| Zoom In | | ↑ |
| Zoom Out | | ↓ |
| Zoom Reset | | Ctrl + Del |
| Zoom Center | | Ctrl + Ins |
| Quick View | | Ctrl + Alt + P |

## Time Controls

| Function | Icon | Shortcut |
|---|---|---|
| Previous Clip | | Ctrl + ← |
| Next Clip | | Ctrl + → |
| Set In | [ | I |
| Set Out | ] | O |
| Go to In | n/a | Ctrol + Shift + I |
| Go to Out | n/a | Ctrl + Shift + O |
| Clear In | n/a | Shift + I |
| Clear Out | n/a | Shift + O |
| Clear In&Out | n/a | Ctrl + Shift + Delete |
| Previous Frame | | ← |
| Play Backwards | | → |
| Stop | | J |
| Play Forwards | | K |
| Next Frame | | L |
| Go to Head | n/a | Ctrl + Home |
| Go to Tail | n/a | Ctrl + End |
| Next Keyframe | | Shift + → |
| Previous Keyframe | | Shift + ← |

## On Screen Manipulators

| Function | Icon | Pick On |
|---|---|---|
| Re-Align Stitch | Ctrl + | Main Workspace |
| Rotate Stitch | Alt + | Main Workspace |
| Edge Point Size | Shift + | Edge Point Center |
| Circle Mask Size | Alt + | One Input Workspace |

## Parameter Manipulators

### Sliders

| Action | Result |
|---|---|
| | 1:1 speed |
| Alt + | 1:10 speed |
| Alt + Ctrl + | 1:100 speed |
| Alt + Shift+ | 1:1000 speed |

## Parameter Manipulators

### Numeric Fields

| Action | Result |
|---|---|
| or | 1 Step |
| Alt + or | 0.1 Step |
| Alt + Ctrl + or | 0.01 Step |
| Alt + Shift + or | 0.001 Step |
| Shift + or | 10 Step |
| Ctrl + or | 100 Step |

SGO