

# VR we are

## User Manual



# Content

<b>1</b>	<b>Introduction .....</b>	<b>4</b>
<b>2</b>	<b>Getting Started.....</b>	<b>5</b>
2.1	System Requirements .....	5
2.2	Installation Step .....	6
2.2.1	Install Tools .....	7
2.2.2	Install ComfyUI .....	7
2.2.2.1	Choose Installation Option .....	7
2.2.2.1.1	Option 1: Windows Portable Package .....	7
2.2.2.1.2	Option 2: ComfyUI Desktop .....	7
2.2.3	Install Prerequisites on ComfyUI .....	7
2.2.3.1	Install ComfyUI Custom Nodes .....	7
2.2.3.2	Install ComfyUI Models.....	8
2.2.3.2.1	Upscale Model .....	8
2.2.3.2.2	Depth Model .....	8
2.2.3.2.3	Dubbing Model .....	8
2.2.4	Install „VR we are“ .....	8
2.2.5	Configure „VR we are“ .....	9
2.2.5.1	Configure for ComfyUI Desktop.....	9
2.3	Setup of the VR headset .....	10
2.4	Skybox VR Player.....	10
2.5	DEO VR Player.....	10
2.6	4XVR Video Player.....	10
2.7	potplayer.....	10
<b>3</b>	<b>System Overview .....</b>	<b>11</b>
3.1	User Interface .....	12
3.2	Tool Pipeline.....	12
3.3	Cleanup .....	13
<b>4</b>	<b>Stages.....</b>	<b>14</b>
4.1	Caption.....	14
4.2	scaling.....	14
4.3	fullsbs.....	14
4.4	interpolate .....	14
4.5	singleloop.....	14
4.6	dubbing/sfx.....	14
4.7	slides .....	15
4.8	slideshow .....	15
4.9	watermark/encrypt.....	15
4.10	watermark/decrypt.....	15
4.11	concat .....	15
<b>5</b>	<b>Tasks .....</b>	<b>16</b>

5.1	<i>vlimit-2160p, vlimit-1080p and vlimit-720p</i> .....	16
5.2	<i>hlimit-4k</i> .....	16
5.3	<i>fpslimit-30, fpslimit-15</i> .....	16
5.4	<i>reencode</i> .....	16
6	<b>Customization</b> .....	17
7	<b>Advanced Features</b> .....	18
7.1	<i>Topaz Video AI</i> .....	18
8	<b>Troubleshooting</b> .....	19
9	<b>Acknowledgements</b> .....	20



## 1 Introduction

Welcome to „VR we are“, an immersive productivity and creativity software designed to bring the familiar multimedia content of videos and images into the world of a Virtual Reality (VR) environment, using Artificial Intelligence (AI) models on the local computer, and most of the processing can be done offline.

Key feature is conversion of 2D images and videos into full stereo side-by-side (SBS) left-right (LR), which can be visualized with viewers in VR headsets or glasses on 3D-capable TV displays.

Some other features can be used even without targeting a VR device, such as capture meta data, upscaling, frame interpolation, various ffmpeg tasks, dubbing or creating slideshow videos from images.

A central principle of the application is to use conventions over configuration. That said I try to keep *required* configuration to a minimum by providing default settings wherever possible.

## 2 Getting Started

This chapter explains the basic system requirements, installation steps, and setup of the VR headset.

### 2.1 *System Requirements*

The AI tools require a modern computer to perform well. Nevertheless, small media files can be handled on older hardware.

Operating System: Windows 10 or higher.

RAM: 16 GB minimum, 64 GB or higher recommended.

VRAM: Not required. 16 GB or higher recommended.

GPU: Not required. CUDA-capable GPU and setup recommended.

File system: 40 GB free space minimum, 200 GB or more recommended. More than 10 GB free space is required to run.



## 2.2 Installation Step

The installation goes in the following order:

1. Install Tools
2. Install ComfyUI
3. Install Prerequisites on ComfyUI
4. Install „VR we are“
5. Configure „VR we are“



## 2.2.1 Install Tools

If you already have them, I recommend updating, since old version may cause problems.

- Exiftool. Rename exe (strip -k) for usage in CLI and add to PATH environment variable, it will be detected later, and simplifies setup.  
<https://exiftool.org/>
- FFmpeg. Add to PATH environment variable, it simplifies setup.  
<https://ffmpeg.org/>
- Git Bash (for Windows).  
<https://gitforwindows.org/>
- Topaz Video AI (TVAI) – You need current version (6); future versions will stop supporting CLI. You must configure paths later.

Change setup after installation, by editing the following variables in config.ini: EXIFTOOLBINARY, FFMPEGPATHPREFIX, TVAI\_BIN\_DIR, TVAI\_MODEL\_DATA\_DIR and TVAI\_MODEL\_DIR.

## 2.2.2 Install ComfyUI

### 2.2.2.1 Choose Installation Option

#### 2.2.2.1.1 Option 1: Windows Portable Package

Download the 7z file from the latest release on GitHub:

<https://github.com/comfyanonymous/ComfyUI>

It includes Python. Execute run\_cpu.bat or the nvidia ones if you have that GPU.

#### 2.2.2.1.2 Option 2: ComfyUI Desktop

You need to install Python 3.12+ and ComfyUI Desktop:

<https://www.python.org/downloads/>

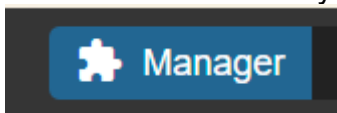
<https://www.comfy.org/>

## 2.2.3 Install Prerequisites on ComfyUI

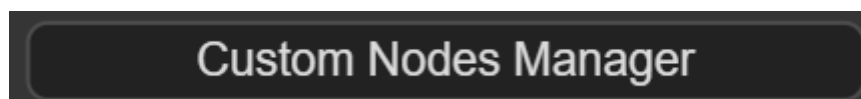
### 2.2.3.1 Install ComfyUI Custom Nodes

In ComfyUI, if not already there, install the Manager. (Download "Code" ZIP and move folder into ComfyUI\_windows\_portable\ComfyUI\custom\_nodes.

Then restart the ComfyUI server and you will see this icon:



Now wait for the Manager to update the ComfyRegistry (watch logs), so the custom node database is up to date.



Now, you can use the ComfyUI Manager to install other custom node packs. It will download most of them from GitHub:

- comfyui\_controlnet\_aux 1.1.0
- ComfyUI-Custom-Scripts 1.2.5
- comfy-mtb 0.5.4
- ComfyUI-Crystools 1.26.8
- ComfyUI-Florence2 1.0.6
- ComfyUI-VideoHelperSuite 1.7.4
- ComfyUI-Frame-Interpolation 1.0.7

Then install ComfyUI-MMAudio manually:

Download from <https://github.com/kijai/ComfyUI-MMAudio>, then install to ComfyUI/custom\_nodes folder, then execute in the bash shell:

```
# Locate and switch to installation folder:
cd ComfyUI_windows_portable
# Execute install script
./python_embedded/python.exe -m pip install -r ComfyUI/custom_nodes/ComfyUI-MMAudio/requirements.txt
# create model directory
mkdir -p ComfyUI/models/mmaudio
```

## 2.2.3.2 Install ComfyUI Models

### 2.2.3.2.1 Upscale Model

Download 4x-realesrgan-x4plus manually from <https://openmodeldb.info/models/4x-realesrgan-x4plus> and place it at ComfyUI\models\upscale\_models\RealESRGAN\_x4plus.pth  
Or use and configure other upscale models in config.ini later.

### 2.2.3.2.2 Depth Model

With version 3.0 depth\_anything\_v2\_vitb.pth is preconfigured, since the giant model is a performance killer. If you want to use it anyway then download Depth-Anything-V2-Giant manually from [https://huggingface.co/likeabruh/depth\\_anything\\_v2\\_vitg/tree/main](https://huggingface.co/likeabruh/depth_anything_v2_vitg/tree/main) and place it at ComfyUI\custom\_nodes\comfyui\_controlnet\_aux\ckpt\depth-anything\Depth-Anything-V2-Giant\depth\_anything\_v2\_vitg.pth and configure it later in config.ini

### 2.2.3.2.3 Dubbing Model

Please follow this Guide on Github to install MMAudio models:

<https://github.com/kijai/ComfyUI-MMAudio?tab=readme-ov-file#installation>

In short:

Download the fp16 variants of the models from huggingface (4 files):

[https://huggingface.co/Kijai/MMAudio\\_safetensors/tree/main](https://huggingface.co/Kijai/MMAudio_safetensors/tree/main)

and move them to ComfyUI/models/mmaudio

## 2.2.4 Install „VR we are“

Use ComfyUI Manager to install the latest version of „VR we are“ (= comfyui\_stereoscopic).  
When restarting the ComfyUI server, watch the log files for any errors.

From the ComfyUI/custom\_nodes/comfyui\_stereoscopic folder you can create desktop shortcuts from the two batch files, by dragging them with the right(!) mouse key on the desktop. They will call shell scripts. Instead you can still call the shell scripts by hand from git bash; this can be necessary to investigate errors. The desktop shortcuts can be configured with icon and you can choose to minimize the window shell.



Now call `demon.bat` or `daemon.sh` and the installation will complete and be tested (takes up to 3 minutes). You will notice a small guy window to popup as well. If the tests went well, you can do the configuration. Close the daemon by pressing CTRL + C in the bash shell.

To call the script manually, open a new Git Bash Shell, and drag the script there or enter:

```
# Locate and switch to installation folder:
cd ComfyUI windows portable/ComfyUI
# Execute background task
./custom_nodes/comfyui_stereoscopic/daemon.sh
```

## 2.2.5 Configure „VR we are“

The properties are located at `ComfyUI\user\default\comfyui_stereoscopic\config.ini`. I recommend to complete and check and complete the values for the following keys: `EXIFTOOLBINARY`, `DESCRIPTION_LOCALE`, `ISO_639_2_CODE`, `WATERMARK_LABEL`

If TVAI is installed provide `TVAI_BIN_DIR`, `TVAI_MODEL_DATA_DIR`, `TVAI_MODEL_DIR` (request CLI instructions Video AI - Nvidia).

If you don't want to the pipeline, e.g. it is confusing you; disable auto forwarding by setting `PIPELINE_AUTOFORWARD` to 0. Auto forwarding is a powerful feature; you can edit the pipeline in `ComfyUI\user\default\comfyui_stereoscopic\rebuild_autoforward.sh` which is executed every time the daemon is started.

Many more properties can be modified there. If the tool fails, delete the config file and the generator will generate a new one with default values.

You need to restart `daemon.sh` after changing the configuration. You do not need to restart the ComfyUI server.

### 2.2.5.1 Configure for ComfyUI Desktop

For ComfyUI Desktop, the port needs to be changed, since the scripts are using the default port of ComfyUI Portable. Go either to Settings->Server Config, and set the value for the port to 8188, or change the value of the property `COMFYUIPORT` in `config.ini` to the ComfyUI Desktop port (8000, look into Settings in ComfyUI Window). Sometimes when you close the window, the new instance will allocate a different port, like 8002, because the old process has not terminated.

## **2.3 Setup of the VR headset**

### **2.4 Skybox VR Player**

Commercial application. Supports many platforms. We have tested it on Meta Quest 3.

To simplify the configuration follow their guide: How to Adjust 2D/3D/VR Video Formats, e.g. by putting all videos and images under a folder with the name "fullsbs".

*Known problems:* To view images, the curved mode must be deactivated (bugged). Projection problems with height adjustment at high zoom.

### **2.5 DEO VR Player**

Download for free. Supports many platforms. We have tested it on Meta Quest 3, HTC vive pro 2.

To simplify the configuration follow their guide: Naming convention, e.g. by adding "\_SBS\_LR" to the end of the file name.

*Known Problems:* Projection problems with height adjustment at high zoom. No controller mapping for pitch. Subtitle track not selectable.

### **2.6 4XVR Video Player**

Another Commercial application.

*Known problems:* Auto-detection of aspect ratio currently does not work (support pending). The player does not support images. Manual setup per file required to 3D FSBS if aspect unusual. No height adjustment. Subtitles not supported.

### **2.7 potplayer**

Download for free. Requires an active connection link to the computer. Can be used as video source for other VR media players that run on the Quest 3, such as Bigscreen.

*Known problems:* The player does not support images. Other video aspects than 16:9 are not suitable for SBS display.

### 3 System Overview

The following picture illustrates the building blocks of „VR we are“:



„VR we are“ is using other software as foundation:

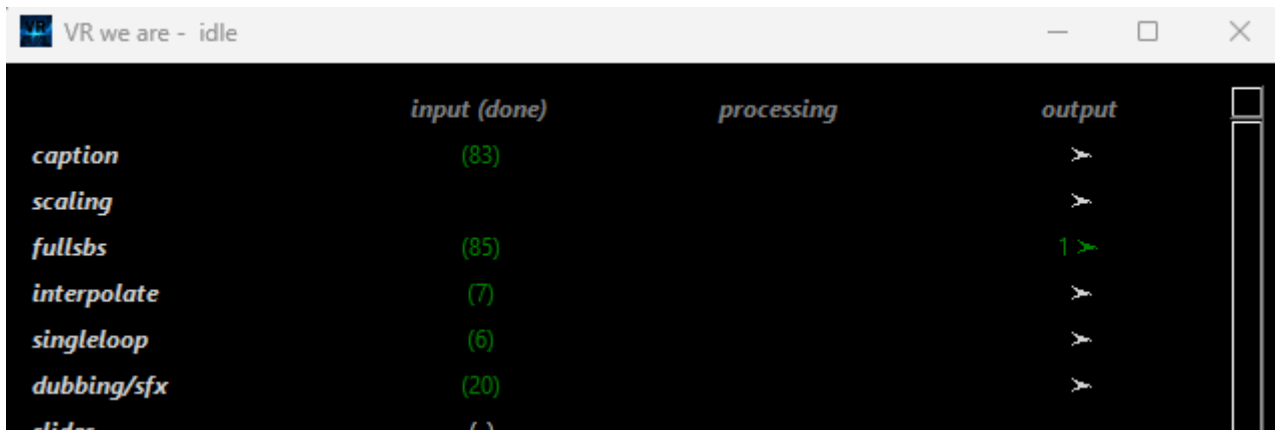
- ComfyUI is building foundational open source software for the visual AI space.  
„VR we are“ uses it as distribution and execution platform
  - Stereoscopic is a custom node package for ComfyUI containing the „VR we are“ software. For the custom node I got help from iablunoshka, responsible for high performance of the SBS converter. Our first tests had been made with the nodes of SamSeen.
- FFmpeg is a command line tool providing a multimedia framework for video and image manipulation.
- Exiftool is a command line tool for reading and editing multimedia meta information.
  - Google Trans is an optional service, requiring to be online, to translate text (into the own locale).
- Topaz Video AI (TVAI) is an optional professional product, when available used. It offers a massive speed and quality boost on scaling and video interpolation (frame rate increase).
- Git Bash package, an application for Microsoft Windows environments, which provides an emulation layer for a Git command line experience, required to execute „VR we are“.

„VR we are“ waits for multimedia files to be placed in input funnels (file folders) for processing. Per default the files are processed in a non-linear pipeline, landing in output baskets (file folders). Pipelining though the stages can be customized or even completely turned off.

**Important: Never place your original files there; only copies!**

### 3.1 User Interface

A small graphical user interface is started with the daemon. For now the only operation is showing the status with a table view of the stages:



	<i>input (done)</i>	<i>processing</i>	<i>output</i>
<i>caption</i>	(83)		✓
<i>scaling</i>			✓
<i>fullsbs</i>	(85)		1 ✓
<i>interpolate</i>	(7)		✓
<i>singleloop</i>	(6)		✓
<i>dubbing/sfx</i>	(20)		✓
<i>slides</i>	(-)		

### 3.2 Tool Pipeline

To simplify things, the tool offers a video and image processing pipeline with auto-forward from stage output to another stage input. With release 3.0 it is no longer linear. It can be customized with care.

You just need to place files in input folder of a stage, and pick the results up from the final output folders. No more need for common users to create own workflows or other CLI command and shell scripts.

If you get confused by auto-forwarding, just disable it by setting in config.ini  
PIPELINE\_AUTOFORWARD=0

The transfers from output to input folder of stages is automatic and defined in ComfyUI\user\default\comfyui\_stereoscopic\rebuild\_autoforward.sh. This shell script is called on every start of the daemon. It contains lines of the following format:

```
echo '[CONDITIONS]INPUTSTAGEPATH' >>output/OUTPUTSTAGEPATH/forward.txt
```

e.g.

```
echo '[sbs=false:height>2160]tasks/vlimit-2160p' >>output/vr/scaling/forward.txt
```

These lines are copied into *forward.txt* files in the output stages. Media files that land there will be checked against these rules for transfer from output to input stage. The rules are executed in order and the first rule that matches is executed.

Conditions are rule details, which must all match for the rule to be executed. Conditions are optional. There is an invisible rule that is always checked: rules are only executed if the media type (video or image) is supported by input and output stage. Multiple conditions are defined by using the colon separator. The script contains more information about rules in the header.

### ***3.3 Cleanup***

Processed files in input folders are copied into a done subfolder causing a lot of file copies. These folders are cleared when the daemon is restarted. At installation time (once) *.nocleanup* files are created in this folder. If they exist, the files are excluded from cleanup. You can create and delete this *.nocleanup* files as you need. This system is one of the reasons you should never place your original files in input folders. Another reason is, that the application is renaming files. Try to avoid using filenames with non alpha-numeric characters [A-Za-z0-9], especially if they come from untrustworthy sources. I do some checks but can't guarantee it is bullet proof.

## 4 Stages

The following stages are the basic operation „VR we are“ can perform:

### 4.1 *Caption*

Adds title, description to videos and images using analysis of first frame. Translates descriptions if online. They are displayed and editable in the Windows Explorer in Windows 11 (only displayed in Windows 10). Forwards meta data of videos and images through stages , strips off prompts and workflows. To enable, change language or turn off translation please edit the config.ini parameters EXIFTOOLBINARY, DESCRIPTION\_LOCALE, ISO\_639\_2\_CODE.

### 4.2 *scaling*

Scales the video and image up, using a combination of upscale-model and blending with the original input. High resolution input is required by the SBS converter to produce good results. For maximum quality and if over a minute place them in subfolder "override". Restricted use on old hardware by pipeline conditions. For low quality input and long videos I recommend using tools first. For videos it operates on frames, why fps-rate is limited in the pipeline first. Important config.ini parameters: UPSCALEMODELx4, UPSCALEMODELx2. TVAI requires model configuration.

### 4.3 *fullsbs*

Convert video and image to side-by-side left right, full width to be used in a viewer on your VR display. For best results input should have resolution of at least 1MP or higher. For videos it operates on frames, why fps-rate is limited in the pipeline first.

Important config.ini parameters: SBS\_DEPTH\_SCALE, SBS\_DEPTH\_OFFSET, SBS\_DEPTH\_BLUR\_RADIUS\_VIDEO, SBS\_DEPTH\_BLUR\_RADIUS\_IMAGE, DEPTH\_MODEL\_CKPT

### 4.4 *interpolate*

Interpolate doubles the video frame rate. It is very time-consuming operation. config.ini parameters: TVAI requires model configuration.

### 4.5 *singleloop*

Appends a reverse version to a video, so it can be looped. Useful for short videos that have no forward direction in time. I recommend using this before dubbing videos.

### 4.6 *dubbing/sfx*

Designed to add sound (effects) to videos. Additional prompts can be added. Be aware, that this might be out-of-sync and contain strange parts. It is a tool for you to play with and quickly get some sound. Prompts are injected from ComfyUI\user\default\comfyui\_stereoscopic. Use dubbing before scaling and fullsbs.

Seeds are random. Repetitive calls (if auto-forward is deactivated) may produce new sound results, with ascending numbers, for the user to choose.

Long videos >20 seconds, especially slideshows, generate sound changes every 5 seconds, so every slide has its own sound.



#### ***4.7 slides***

Want to see your images, maybe from vacation, in VR, but are annoyed about the way your VR viewer is displaying them? Use this to create 4K slide images (upscale + padding) as preparation to make a video.,

#### ***4.8 slideshow***

put slides into a slide show, 5s each image including a 1s VR-compatible transition.

#### ***4.9 watermark/encrypt***

encode forensic watermark on images, and keep the original image in store.

#### ***4.10 watermark/decrypt***

decode forensic watermarking from image against stored version and return the watermark image

#### ***4.11 concat***

Can be used in combinations with tasks/split-1m or to put multiple single loop videos together, which share the same first (and last) frame to a longer video. I recommend using this before dubbing.

## 5 Tasks

Tasks are standardized stages based on a blueprint. The user can duplicate and customize them. He can also replace them by modifying the pipeline.

### ***5.1 vlimit-2160p, vlimit-1080p and vlimit-720p***

Ensures the height of a media is limited for performance reasons.

### ***5.2 hlimit-4k***

Ensures the width of a media is limited for publication reasons.

### ***5.3 fpslimit-30, fpslimit-15***

Ensures the frame rate of a video is limited for performance reasons.

### ***5.4 reencode***

Ensures the media is well defined for processing.

## 6 Customization

Users can create their own tasks. For now, the blueprint ffmpeg-v2v can be used. You need to have sufficient knowledge of FFmpeg to do.

Just copy a template from ComfyUI\custom\_nodes\comfyui\_stereoscopic\config\tasks to ComfyUI\user\default\comfyui\_stereoscopic\tasks

E.g. hlimit-4k.json and give it an appropriate name.

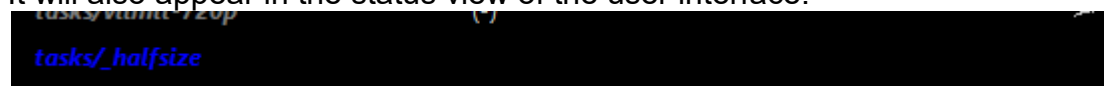
Example: We want to scale videos down by factor 2. We name it halfsize.json.

To produce what we want we change filter in the json file, so the file looks like that:

```
{
  "version": "1",
  "blueprint": "ffmpeg-v2v",
  "options": "-filter:v scale=iw/2:-2",
  "input": "video",
  "output": "video",
}
```

Now restart the daemon and a new stage appears. User stages have an underscore appended to the task name, so the input folder is ComfyUI\input\vr\tasks\\_halfsize

It will also appear in the status view of the user interface:



tasks/\_halfsize

It is not integrated in pipeline, unless you do so. Just drop files in the input folder and see how they are processed.

## 7 Advanced Features

### 7.1 *Topaz Video AI*

Using this software for upscaling and interpolation instead of using ComfyUI models and workflows archives much better results; it is at least 10 times faster and you can perfectly adjust parameters in a preview mode.

Currently tested with v6.0.4, models: prob-4, iris-2, chf-3



## 8 Troubleshooting

Problems or questions? Contact the Discord Support! Activation Link:  
<https://discord.gg/ZegT6Cc8FG>



## 9 Acknowledgements

The software was developed by me, Fortuna, with lot of help from lablunoshka, who is responsible for the insane speed of the SBS algorithm. We did this in the hope, that we soon see more VR content, from AI here or classic somewhere else.

Additional help come from some artists, namely (in alphabetic order): Robotka, schmede, VisionaryAI\_Studio and z1000k. You will see the result of this tool in the many examples in the gallery on Civitai. Please visit and rate their original work (links included in posts), instead of the stereoscopic variant. I want to say "thank you!" to them for the opportunity to propagate the potential of this tool through their artwork.