# 

Holovibes

User manual

# 

# 

# Table of contents

[Table of contents](#_4fj0uodav9pn)

[Introduction](#_8gfja8r3b84i)

[Requirements](#_mrhipbfje7p5)

[User interface](#_wxxgyl1qodlt)

[Menu items and tabs](#_hk8l3jnithsg)

[List of menu items and tabs](#_r7dekownktth)

[Menu bar](#_mg6n8hkv7gih)

[Image rendering tab](#_qwhinwwiflve)

[View tab](#_kdtengz3qq65)

[Post-processing tab](#_j8ti0toa9776)

[Hologram ratio](#_v3be5wr4m0zr)

[Flowgraphy](#_39gcv2gxxl1b)

[Speckle contrast](#_926xke9dmts4)

[Convolution by a user-defined kernel](#_1osy1eg5irma)

[Signal averaging in a region of interest](#_isory3upgvpg)

[Record tab](#_3vujw1z3vtw9)

[Recording of processed image stacks](#_h39h0wmszuzs)

[Recording of spatially-averaged signals](#_tm2kla5d61l3)

[Batch recording of image stacks and averaged signals](#_240vk3x4yqwd)

[Import tab](#_d0sou9wykt8l)

[Info tab](#_hj16kg71bpc3)

[Configuring Holovibes](#_a4c6wgfpnlae)

[Config](#_cyyhapog85z6)

[Image rendering](#_bpwqnjzchzu)

[View](#_qyoyz1beszsq)

[Post-processing](#_kzo619n3zd4y)

[Record](#_9w08wavyf6sq)

[Import](#_ty5b70fzt9od)

[Info](#_16c6arfn0cj9)

[Autofocus](#_dy63vjqdgwt4)

# 

# 

# Introduction

Holovibes is a free software dedicated to the calculation of digital holograms in real-time. It has been designed to accelerate the transition to high throughput digital optics. It turns interferograms into beautiful holograms. Input data can be grabbed from a digital camera or loaded from files recorded beforehand. Massive amounts of data can be handled robustly at high throughput, saved to disk, and visualized in real-time without any risk of frame dropping thanks to the use of several configurable input and output memory buffers. This software is developed in C++ and CUDA on windows 7 (64-bit) and Windows 10 (64-bit) operating systems. It uses NVidia graphics processing units (GPU) for parallel computation.

The main features include:

● Real-time image acquisition from several digital cameras

● Importation of massive interferogram or hologram datasets

● Choice of hologram rendering method

● Real-time hologram rendering and video output

● Real-time temporal short-time Fourier transform analysis

● Choice of output type: phase, magnitude, complex, composite color

● Hologram autofocus

● Dynamic plots of spatially-averaged signals

● Real-time image and video post-processing

● High throughput saving to disc of massive datasets

● Batch recording and communication with instruments via GPIB

Use-case examples:

● [Holographic microscopy](http://holovibes.com/)

● [Holographic OCT](https://youtu.be/mqlxnT32n9k)

● [Holographic vibrometry](https://youtu.be/LYtISSHNcoo)

● [Holographic angiography](https://youtu.be/RhVPXBnPhXc)

● [Holographic plethysmography](https://youtu.be/kGPWyUIM1OA)

This software has been developed and maintained by several generations of students from ESME Sudria, EPITA, and school 42, trained at the Langevin institute - CNRS UMR 7587 in Paris, France.

# Requirements

In order to use the version 5.5 of Holovibes, you will need:

● A computer with at least 4 GB of RAM,

● Microsoft Windows [7](https://www.microsoft.com/en-us/software-download/windows7)/[10](https://www.microsoft.com/fr-fr/software-download/windows10) 64-bit operating system

● A NVidia graphics card (GeForce GTX 700+ series)

● [NVidia CUDA 9](https://developer.nvidia.com/cuhttps://developer.nvidia.com/cuda-downloadsda-downloads)

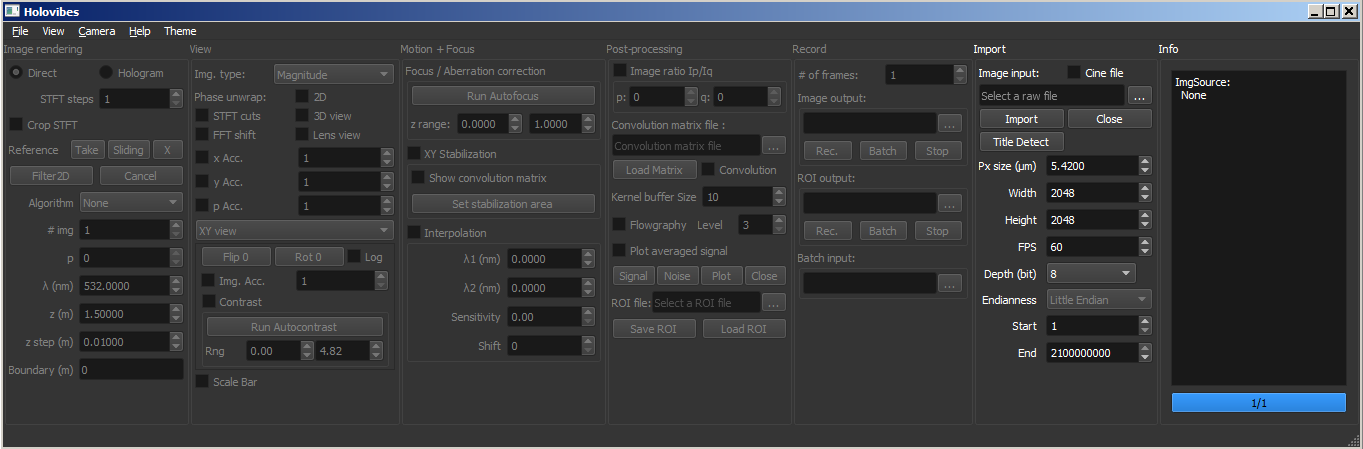
● A supported digital camera, or [raw interferogram files](https://ftp.espci.fr/incoming/Atlan/holovibes/data)

# User interface

The main features of Holovibes include : 1- Buffered Image acquisition at high throughput from various digital cameras. 2- Importation of massive datasets of interferograms. 3- Real-time computation of a video flow of holograms from the input interferograms. 4- Real-time temporal analysis by short-time Fourier transform. 4- Real-time autofocus, and other post-processing methods. 5- Real-time, high throughput buffered recording to disk of massive amounts of raw or processed images and holograms.

# Menu items and tabs

## List of menu items and tabs



● [Image rendering] direct image grabbing, temporal demodulation, beating-frequency spectrum analysis, hologram rendering by Fresnel transform or angular spectrum propagation, spatial filtering, reference image substraction.

● [View tab] Holograms are complex-valued. Output types include : magnitude, phase, accumulation of images, contrast control, autofocus.

● [Motion + Focus] Hologram autofocus, lateral stabilization, real-time interferogram rescaling by linear interpolation.

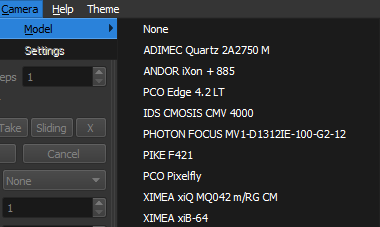
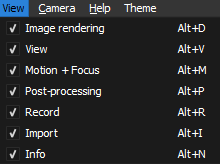
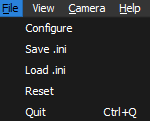
● [Post-processing] Speckle flowgraphy, Spatial speckle contrast, image convolution by a user-defined kernel, Image ratio, signal averaging in a region of interest.

● [Record tab] Displayed images are temporarily buffered in a configurable output queue used to prevent frame dropping while saving.

● [Import] Buffered image grabbing from a camera or from a file imported into an CPU input queue.

● [Info] Input/Output buffers and throughput monitoring

## Menu bar



In the menu bar, you can select the camera to use, and open the main and camera-specific configuration (.ini) files. In the camera menu, you may choose the camera to use. You can also open camera-specific .ini configuration files. If you modify the configuration file and want changes to take effect, select your camera again in the menu bar [camera/model]. In File, you can open the *holovibes.ini* configuration file and exit the program. More information on configuration options is available in the section *Configuring Holovibes*. In View, you can activate full screen mode for the display window (press Ctrl-F or Escape to exit full screen mode), and show/hide the individual panels. You can also use keyboard shortcuts:

● Full screen : Ctrl-F (exit : Ctrl-F or ESC)

● Image rendering : Alt-D

● View : Alt-V

● Post-processing : Alt-S

● Record : Alt-R

● Import : Alt-I

● Info : Alt-N

## Image rendering tab

This panel regroups all the parameters related to the real-time hologram rendering process. You may modify numerical values by clicking on the corresponding arrow signs. At the top, you may choose between Direct (display the last grabbed image), Demodulation (temporal demodulation by linear combination of the latest images), and Hologram (hologram rendering). The integer #img is the number of images on which temporal demodulation is performed. The integer p is the index of the temporal Fourier component of the last #img frames to display. The p index value cannot exceed #img. The lambda value, in nanometers, is the wavelength of the light used to record interferograms. The z parameter, in meters, is the reconstruction distance of the hologram, if the reference used to record the interferogram is a plane wave on the sensor. It can be modified anytime during image acquisition. The z distance may be positive, negative, or null. The value used to increase/decrease the z distance when pressing the up/down arrows of the keyboard is z step, in meters. The algorithm menu selects the rendering method : FFT-1 computes [Fresnel transforms](https://en.wikipedia.org/wiki/Fresnel_diffraction). It is suited to the reconstruction of objects observed far from the sensor, for z greater than the value of the boundary field, FFT-2 performs hologram rendering by [propagation of the angular spectrum](https://en.wikipedia.org/wiki/Angular_spectrum_method) of recorded interferograms. It is suited to objects observed near the sensor, for z lower than the value of the boundary field. Image zooming can simply be performed at any time by zone selection on the image display window. A green box delimiting the zoom field will appear on left mouse click, and the display will revert to full image extent upon right click.

## View tab

This panel lets you tailor the display method to your needs.

● Image type sets the screen display type (which also sets recording type):

● Magnitude : modulus of the complex data.

● Squared magnitude : modulus taken to its square value.

● Argument : phase (angle) value of the complex-valued pixels.

● Phase increase : phase variation from the last rendered hologram to the current one.

● Complex : color display for real and imaginary parts of the complex-valued pixels.

● Composite : color representation of hyperspectral holograms.

● STFT cuts creates two windows XZ and YZ displaying the short-time temporal Fourier transforms (spectrograms) of the signal at each pixel along horizontal and vertical cut lines in red, which can be locked by a spacebar keystroke.

● x Acc, y Acc, and p Acc define the number of pixels over which YZ, XZ, and XY displays are averaged spatially, respectively.

● *Img. Acc.* defines the number of holograms accumulated in time, rendered with any display method. It defines how many past images shall be cumulated with the latest one.

● *unwrap phase* is related to the use of phase increase mode. Activating it adds an extra step to the computation : unwrapping the phase angles obtained, before cumulating them.

● log : sets the log (base 10) filter for the output, at the end of the processing.

● *FFT-shift* option shifts zero-frequency components to the center of the spectrum. Unchecking it does the reverse and shifts back the image corners.

● contrast option allows to adjust the image’s contrast, using the values defined in neighbouring *min* and *max* boxes (*Rng.*). Click on the Auto button to let the program find the best values of min and max by itself.

● Autofocus : lets you select a yellow area on the hologram display window, and the program tries to guess the reconstruction distance z for optimal image focusing. First, select lower and upper bounds for the z distance to search using z min and z max. Then choose the way the algorithm sweeps z values by tuning steps and loops. Steps indicates how many linearly-spaced values between z min and z max will be tried out at each step. Loops indicates how many iterations will be tried out for each step. For instance, using steps = 100 and loops = 1, we tell the program to try 100 different values, from z min to z max. Using steps = 10 and loops = 3, we tell it to proceed in three steps. At each step, the range of reconstruction distances to sweep is narrowed. This second method is usually faster and more efficient, unless the contrast function forms a subsampled sharp peak.

● Lens view : displays the complex-valued lens used for image rendering

● XY, XZ, YZ view : activates the currently selected view for image transformation, contrast, and logarithmic display.

## Post-processing tab

This panel offers several post-processing functionalities : Hologram ratio for frequency-division multiplexing, flowgraphy, speckle contrast, convolution by a user-defined kernel, and signal averaging.

### Hologram ratio

The image ratio option performs complex-data division of images extracted from the latest hologram. It uses the p and q indexes to calculate the image ratio H\_p/H\_q.

### Flowgraphy

The speckle flowgraphy filter computes the sums and the absolute differences of pixel values over the closest neighbours of a given pixel, in space and time, and forms the ratio between the averaged sums and the averaged differences.

### Speckle contrast

The speckle contrast filter computes the local contrast of the current image, which quantifies local levels of blurring. Under coherent illumination, the level of blurring will differ depending on local movement. The more local motion, the more blurred local speckles will appear.

### Convolution by a user-defined kernel

Image convolution by user-defined spatio-temporal kernels can be performed in real-time. Convolution kernels have to be defined in ascii files and loaded by clicking on Load Matrix, after selecting the path to the convolution matrix file.

### Signal averaging in a region of interest

The ROI (Region Of Interest) averaging functionality allow data selection, dynamic plotting, and saving of signals averaged in a parametrable ROI. Check the Avg. signal in ROI box to enable zone selection on on the image display window. The first pink box will correspond to the signal, and the blue one to the noise, as reminded on the panel. Click the Plot button to pop an extra window showing data plotting in real time, limiting data to the zones selected. If you want to update the pink and/or blue boxes, click again on Plot button to update the real-time plot. The abscissa shows the iterations indexes; the number of iterations plotted can be configured through the Points # box. The ordinate shows values depend on the type of output. You can rescale the Y-axis anytime using the Auto scale button. Finally, use the list box at the right to select the data to be plotted : signal, noise, signal/noise, 10\*log(signal/noise). The last section of the panel enables you to save information on the signal and noise zones selected, or load them from disk. Click on the “...” button to open a file explorer, or type directly the absolute path to the file to open. Click then on Save ROI to save the file on disk, creating it if necessary, or Load ROI to use that file : you should see the signal and noise regions updated on screen.

## Record tab

The contents of this tab are used to configure hologram recording.

### Recording of processed image stacks

The “# of frames” box can set the number of consecutive frames which will be recorded in the output raw file. Name your output file in “Image output” and click on Rec. to acquire the frames. By default, it will export the data into a raw file, and the pixels will be formatted in 8-bit or 16-bit unsigned integers. If the Float output checkbox is checked, the pixels will be formatted as 32-bit floats. If you press Stop during a recording, the operation is stopped, and the amount of data saved in the output file will be readable.

### Recording of spatially-averaged signals

If you put a file location in ROI output and then click on Rec, you will record, in a CSV format file, data grabbed from the ROI zones : the signal, noise, signal/noise, and 10 \* log(signal/noise).

### Batch recording of image stacks and averaged signals

By default only one file will be produced, but if you enter a batch input file in the Batch input line, and then click on Batch, then that file will be read, and it will allow you to record multiple files. This file can also communicate via the GPIB interface with remote instruments such as arbitrary function generators. Let’s take a look at this example batch file. Comments are marked as ‘//’ here to explain what is going on but the current version of HoloVibes doesn’t recognize them so remove them if you want to use that file.

|  |
| --- |
| #Block // Pure legacy, useless, just here for visual separation.  #Capture // Tells holovibes to record # of frames.  #Block  #WAIT 2000 // in ms  #InstrumentAddress 20 // set up connection with GPIB Instrument 20.  FREQ 3 // No ‘#’ here; this command is sent to GPIB instrument 20.  VOLT 1 // then this command is sent to GPIB instrument 20.  #WAIT 2000  #Block  #Capture  #Block |

Note that trying to connect to 2 Instruments and then sending a command will only send it to the last one entered.

|  |
| --- |
| #InstrumentAddress 20  #InstrumentAddress 22  FREQ 3 // Will only send it to instrument 22. |

## Import tab

Image importation enables offline processing of massive raw datasets of interferograms, with real-time visualization. In the Import tab, file reading can be configured; reading from a .raw image file instead of being grabbed from a camera. If checked, the Loop check box forces sequential reading of the same file. When the program has finished reading the file, it starts back again from the beginning. Pressing the Import button will use the image stack from the .raw file specified as image source. Pressing the Close button will stop image acquisition from this file. The Width and Height parameters are the lateral extent of the images that will be read out from the .raw file. Please note that any imported frame will be zero-padded to a square frame whose lateral size will be the nearest power of 2. The FPS (frames per second) parameter configures the reading rate. Some cameras will provide images with a bit depth of 8 whereas on some others they are recorded with a bit depth of 16. This needs to be correctly set in the “Depth (bit)” field. For 16-bit frames, the field Endianness becomes editable, and the user can choose whether imported images should be interpreted in little or big endianness. Finally, the Start and End fields can select how many frames need to be read from the imported image stack.

## Info tab

The info panel gathers and displays information about the real-time state of Holovibes. ImgSource tells what is the current input source that is being used. It can be either a file or a camera. InputQueue displays the current number of frames in the input queue, out of the total buffer capacity. It can be used to know whether or not the software can actually cope with computations (demodulation, convolution, hologram rendering...), because the frames are put in the queue as they are grabbed but are only taken away from the queue at the pace of the required image operations. Hence, if the number on the left is low, you can tell that the program is not being slowed down. However, if the input queue reaches its maximum, newer frames will overwrite old ones, meaning that some frames will be lost while recording. Dropped frames can be avoided by increasing the size of the input buffer in the configuration file holovibes.ini” (see next section). OutputQueue tells how many frames are currently in the output queue, out of the total capacity, waiting to be recorded. During a recording, if the number on the left is increasing, it means that the recording process is getting slow. You may use a faster drive to save your results. Rendering displays the rendering frame rate. Finally, the bottom bar is a loading bar that will display the completion of recording processes. The “x/y” label tells that x images out of y have been recorded to the output file.

# Configuring Holovibes

The main configuration file, “holovibes.ini”, contains various default values for all the control panels of the program. It is accessible via the menu [ File > Configure]. Configuration options are gathered by topic. The configuration file “holovibes.ini” is saved upon closing. Hence you will find the previously used parameters when launching the program.

## Config

Configuration of input and output memory buffers.

● input\_queue\_size : The number of frames of the input queue, used to store images received, either grabbed from a camera or read from a file. A higher limit allows working with faster cameras, but remember that this takes much memory. For example, a buffer of 1024, 4 Megapixel, 16-bit frames will use 8 GB of GPU RAM.

● output\_queue\_size : The size of the output queue used to bufferize unsigned integer (uint16 format) images before display, in GPU RAM. The size of this buffer should be increased in case of dropped frames upon image recording.

● output\_queue\_float\_size : The size of the output queue used to record floating point (32-bit float format) images. The size of this buffer should be increased in case of dropped frames during image recording to disc. For example, a buffer of 1024, 4 Megapixel frames will use 16 GB of GPU RAM.

● frame\_timeout : In milliseconds. The maximum time allowed for the camera to provide a new image.

● flush\_on\_refresh : If set to true, each time a parameter is changed on the control panel, the input queue is emptied before resuming filling it.

● reader\_buf\_size : The number of frames read from disk and copied to RAM at once, when using the Import functionality. A number too low cripples performance.

● accumulation\_size : The number of past images summed up to the current one when using phase 1 or phase 2 image modes.

## Image rendering

Default parameters for hologram computation from a stream of images. These parameters are the ones you can see on the Image rendering panel.

● hidden : If set to true, the Image rendering panel is hidden.

● camera : The index of the camera used. The index starts from 0, at the top of the list displayed when you click on Camera in the menu bar. For instance, 2 corresponds to the PCO Edge 4.2 LT camera.

● # img : The number of images to demodulate temporally.

● p\_index : The index of the temporal Fourier component displayed on screen. Cannot exceed phase number, naturally.

● lambda : The wavelength of the light used to perform hologram rendering.

● z\_distance : The “focal length” : reconstruction distance of the hologram.

● z\_step : The step used to increment/decrement z\_distance.

● algorithm : The processing performed to render an hologram. You may choose between FFT1, FFT2

## View

Default parameters on various post-processing operations. These parameters are the ones that can be modified in the View tab.

● hidden : If set to true, the View panel is hidden.

● view\_mode : The image mode used : you may choose between : magnitude, squared magnitude , argument, phase 1, phase 2.

● log\_scale\_enabled : If set to true, the output will be filtered through the logarithm (base 10) function.

● shift\_corners\_enabled : If set to true, the zero-frequency components of the spatial spectrum will be shifted to the center of the display.

● contrast\_enabled : If set to true, the image contrast will be corrected after having been processed.

● contrast\_min : The lower bound used to apply contrast.

● contrast\_max : The higher bound used to apply contrast.

## Post-processing

Default parameters on the other operations you can find in the Post-processing panel.

● hidden : If set to true, the Post-processing panel is hidden.

● image\_ratio\_enabled : If set to true, the ratio of two demodulated holograms, H\_p and H\_q, is calculated.

● image\_ratio\_q : Index of the temporal Fourier component used as denominator in hologram ratio.

● average\_enabled : If set to true, plotting of the plot window is enabled with selected signal and noise areas.

## Record

hidden : If set to true, the Record panel is hidden.

## Import

hidden : If set to true, the Import panel is hidden.

## Info

hidden : If set to true, the Info panel is hidden.

## Autofocus

size : lateral size of the square matrix used for local variance computation.

z\_min: Lower boundary used to seek z

z\_max: Upper boundary used to seek z

steps: Number of steps used to seek z between z\_min and z\_max.

loops: Number of iterations for z-seeking on a refined range