
59
37
[ARRAST_VJ]
FREE SOFTWARE FOR AUDIOVISUAL CREATION
User Manual
91
55
75

[ARRAST_VJ]

FREE SOFTWARE FOR AUDIOVISUAL CREATION

v. 1.0

User Manual

ABOUT THIS GUIDE

This is the first version of User's Manual as a reference to installation, configuration and use guidance of the [ARRAST_VJ].

By creating this material we intend to support workshops and independent studies about this software. We express the will to collaborate with a growing field of production and audiovisual learning in which free and open source tools are developed, shared and improved - collectively.

The interface documented in this guide, its window usage conventions, mouse clicks, examples and directory structure works under the free operating system GNU / Linux (Ubuntu-MATE 16.10) according to that principle.

However, we described the installation procedures in main operating systems and tried to demonstrate step-by-step every feature - so that the guide can be followed without loss of information by those using different operating systems, different distributions and graphical working environments.

SOFTWARE VERSION

[ARRAST_VJ] v.1.0 - GNU/Linux and MacOS
[ARRAST_VJ] Light - no audio engine (Windows, GNU/Linux, MacOS)

You check the current software and user's guide version accessing the project website: **arrastvj.org** .

TABLE OF CONTENTS

[ARRAST_VJ]


[ARRAST_VJ]
MIX
MAP
ARRAST_VIDEO
UNIDADES
OPEN_IMAGE
CAMERA
EXPORT

INTRODUCTION

[ARRAST_VJ] is a **free software for audiovisual creation** that enables **real time manipulation** of videoclips (with sound), images and cameras, and also the creation of **interactive compositions which may be stored, reproduced and exported.**

It features an effects input module, resources for mixing (MIX mode) and 2D mapping (MAP mode), apart from an OSC communication interface for merging with other software and hardware, everything in an open source platform.

It is developed with Pure Data and compatible with major operating systems.

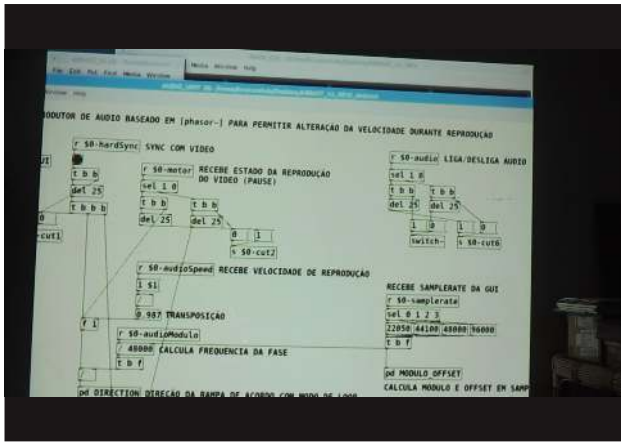
PURE DATA [Pd]

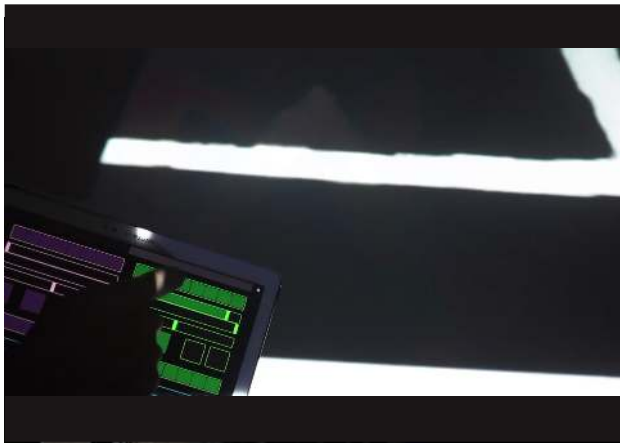
[ARRAST_VJ] is developed and runs from Pure Data programming environment.

Pure Data (http://puredata.info) - or just **Pd** - is a **real time graphical programming environment** originally developed for audio processing but its applications were soon extended to video and computer graphics. The groundwork was developed and maintained by Miller Puckette from 1996 on. Since then, it became a collective project where a large community of developers participates by maintaining and creating new functions that currently cover various fields of artistic and technical applications.

It is a multiplatform tool - it works on any operating system and in the most varied hardware types, from old computers, cell phones and tablets, to cutting-edge computers and video games.

In Pd programming environment, **data from any source** (audio, video, sensors, internet, other software and so on) are treated as being the same thing, that is, **pure data**. It allows creating cross-interactions between all of these materials, which makes Pd an extremely versatile tool with limitless potential for usage in any art field - features that may extend to applications created in this programming environment, such as [ARRAST_VJ].





SET UP

Hardware and software minimum requirements

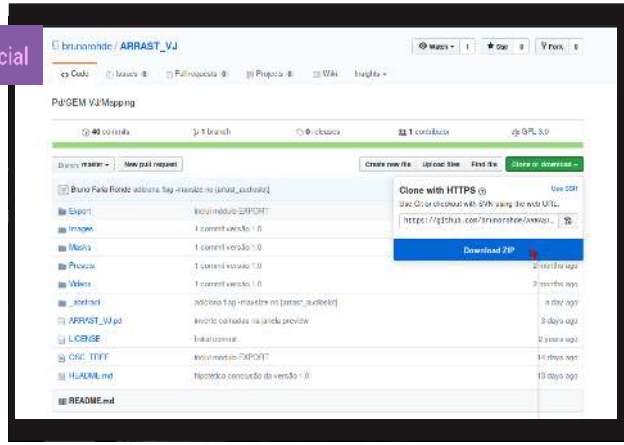
- Dual core processor 1GHz (2 GHz better)
- 1 GB RAM (2 GB better)
- Pd 0.47.1 (Vanilla) or later version
- GEM 0.93.3 (Pd external library)
- Current [ARRAST_VJ] version

Download [ARRAST_VJ]

Direct link for current version is available on

https://github.com/brunorohde/ARRAST_VJ/archive/master.zip

Repositório Oficial



Download this package and extract the zip files to the folder you intend to work.

For more update options, description of single files version or other [ARRAST_VJ] information, visit the project site

ARRASTVJ.ORG

and the development homepage **github.com/brunorohde/ARRAST_VJ**.

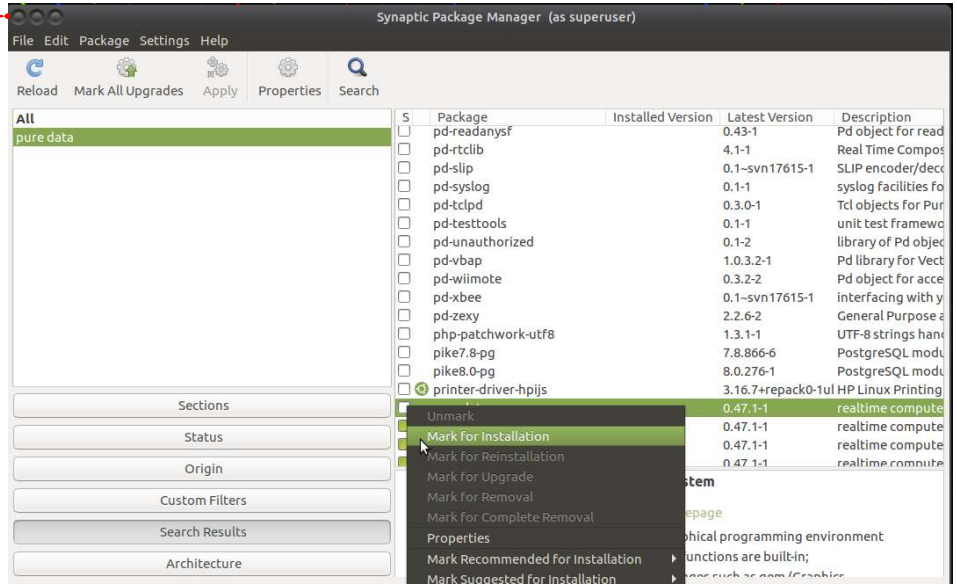
[Pd] + GEM INSTALLATION

For GNU/Linux

If you use Debian, Ubuntu or Mint distributions and similar you may install Pd directly from the repository through **Synaptic** Package Manager or **apt-get** in terminal.

Open Synaptic:

System > Administration > Synaptic Package Manager



Click **Reload** (for updated information about available packages).

On **Search** icon, look fort **pure data**:

1: SELECT **PURE DATA** PACKAGE – AND **"MARK FOR INSTALLATION".**

2: SELECT ALSO **GEM** PACKAGE – AND "T".

3. ON THE NEXT DIALOG, **"MARK ADDITIONAL REQUIRED CHANGES"** AND **APPLY.**

If you prefer installing by command line, open the terminal and type:

sudo apt-get install puredata gem

Follow the terminal instructions to decide about changes and then press **enter** to install.

For other GNU/Linux based distributions look for the best option right on Pd* homepage. If you can't find a direct installation binary, you may download the source code and **compile** the program as follows:

Open **terminal** and browse the folder where you downloaded the Pd code files; type the commands below (one at a time):

```
sudo apt-get build-dep puredata
./autogen.sh
./configure --enable-jack
make
sudo make install
```

Fort Windows ou Mac

Visit **puredata.info** (or the official Miller Puckette site http://msp.ucsd.edu/software.html).

The binaries are ready for installation on the **downloads** section.

For Mac it is **necessary** to install the 32-BIT version even if it is a 64-BIT operating system – because GEM doesn't work with 64-bit Pd in Mac.

Windows - notes:

A strategy to solve GEM dependency problems in Windows is to keep a pd-extended* setup apart from pd-vanilla.

https://puredata.info/downloads/pd-extended

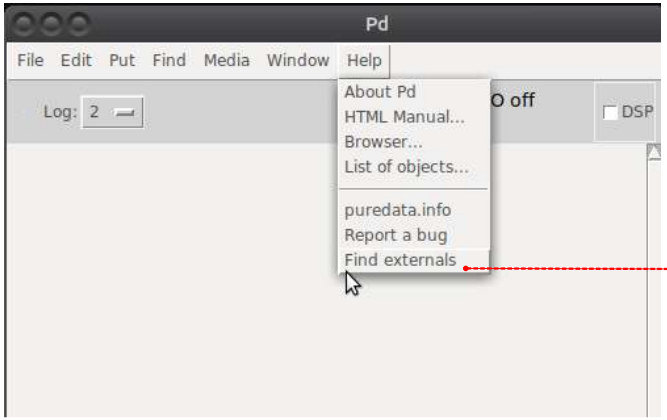
For all codecs and conversion tools: install **Quicktime**, **VLC** and **mpegstreamclip**.



* http://puredata.info/docs/faq/faqsection_view?section=Installing

About GEM

The latest version available is 0.93.3. **If you haven't installed Pd and GEM from the system repository** (as described in the beginning of this section) you must download and install GEM from Pd in the tab **Help > Find Externals > (Type gem in search field) > Search**.



Click the **GEM** option found which comes highlighted in a different colour. The plugin should automatically install the library.

Restart Pure Data after that and [ARRAST_VJ] will run*.

*** KNOWN EXCEPTIONS: GEM CURRENTLY DOESN'T RUN IN 32-BIT GNU/LINUX OPERATING SYSTEM (ONLY 64-BIT) AND IT ALSO DOESN'T RUN IN MAC OSX FROM VERSION 10.11 ON (EL CAPITAN).**

PREPARING THE MATERIAL

What kind of content?

You can work with features from several audiovisual sources ever since they respond to format specifications recommended for each content. Besides, the workflow may be more steady and synchronized as you organize the files you intend to use in your project. Check the videos resolution, the rate of frames per second (fps) and audio samplerate. The following parameters are recommended, according to the content:

VIDEO

Resolution: 640 x 480
Frame rate: 24 fps
Formats: Linux - .avi .mov .mp4
Mac - .mov
Windows - .avi
Codec MJPEG / Motion JPEG
Variable **bitrate**, with an average of 5000 kbps
Suggested **video length**: 30 seconds per clip maximum
File size (MB): variable for each clip depending on chosen settings.

NOTE ABOUT LIMITS:

There isn't real resolution “limit” or videos and images fps limit. Appropriate working limit in each system is given by the settings of each machine.

Computers with less resources may process videos with less quality, less layers, applying less effects – while more powerful computers allow using videos with bigger resolution, applying more effects and so on.

Resolution and fps indications in this guide are an “average” that may be changed according to the background of use. Also, file formats and video codecs depend greatly of the operating system.

SYSTEM AUDIO SETUP:

Operating system: when using audio in [ARRAST_VJ], set the delay in your system as to reduce errors and clicks (bigger delay = less errors)

Pure Data: use 80 msec or more delay and blocksize 512 in Pd audio setup.

AUDIO

An audio file will be loaded along with the corresponding video **only if theses files are in the same folder and have the same name.**

Format: it must always be **.wav**
Samplerate: the audio may be exported in any samplerate (22khz, 44.1khz, 48khz, 96khz), ideally the same used by your operating system and Pure Data audio settings.

IMAGE

Resolution: try working with low resolution images in order not to overload processing – 640 x 480 or other light resolution.

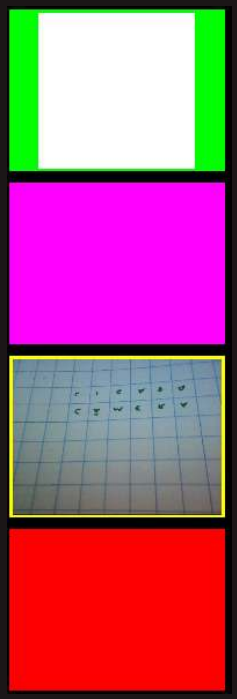
Proportion: any (4:3, 16:9, etc)

Format: always **.jpg** (or .jpeg)

CAMERA

We use by **default** the **laptop's webcam** (if there is one). On Linux other USB devices may also be used, but they have to be **connected before initializing** Pd and then be chosen by its ID (0, 1, 2 and so on) in the main control interface where [CAMERA] is indicated.

Resolution: 640 x 480.



IMPORTANT NOTICE:

THE PREVIEW WINDOW MUST ALWAYS BE OPEN FOR THE CAMERA TO WORK.

ENABLE PREVIEW IN THE [MAIN] WINDOW – THE MAIN CONTROL INTERFACE.

MASK

Alpha masks for maps must be **[.jpg]** or [.jpeg] images, created **in black and white** where hidden areas are black and video areas are white.

Resolution: preferably low (even though the program automatically redimensions masks to the same resolution as the video/image/camera where it is being applied).

Folders / Directories organization

Try saving each kind of content in its corresponding folder according to the file structure of [ARRAST_VJ].

Video and audio (always in the same folder): save with the same name (and different extensions) in ~/ARRAST_VJ/Videos/

Images: ~/ARRAST_VJ/Images/

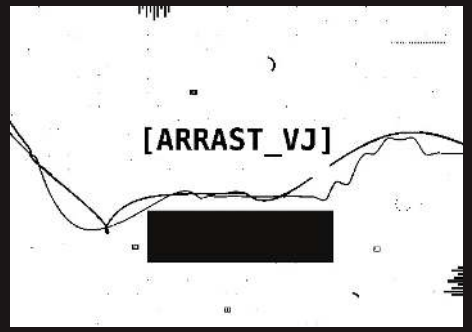
Maps: ~/ARRAST_VJ/Presets/Map/

Masks: ~/ARRAST_VJ/Masks/

Presets: ~/ARRAST_VJ/Presets/Main/

Automations: ~/ARRAST_VJ/Presets/Automation/

1



2

EXAMPLES OF MASKS FILES:

1 BLACK AND WHITE

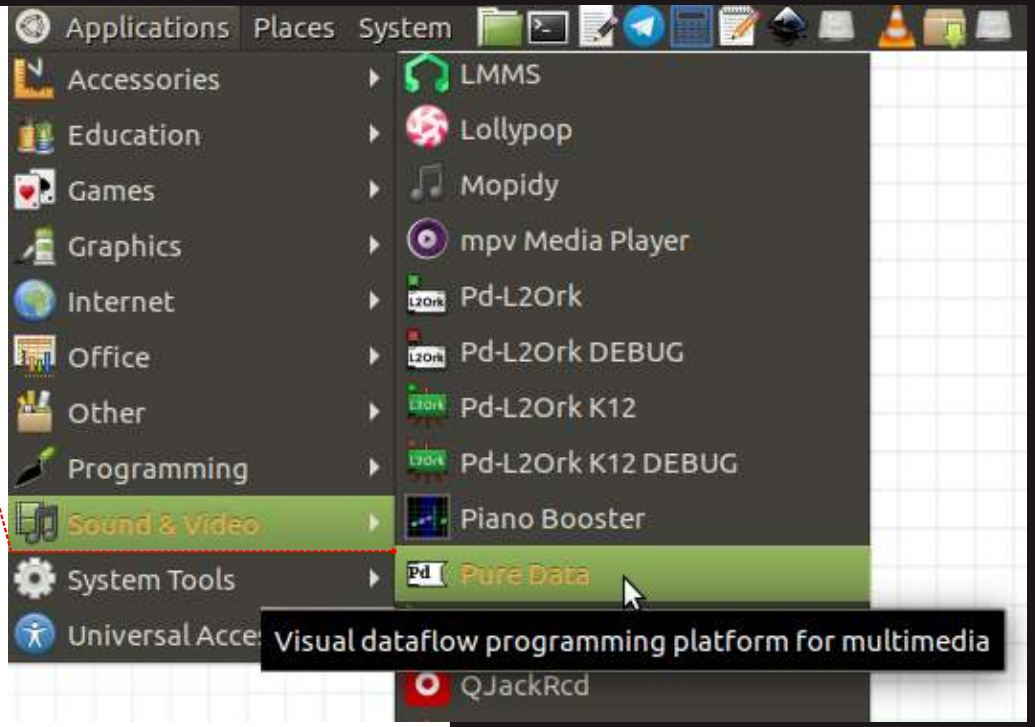
2 GRAYSCALE: TRANSPARENCY AREAS ARE FORMED ACCORDING TO THE GRAY LEVEL

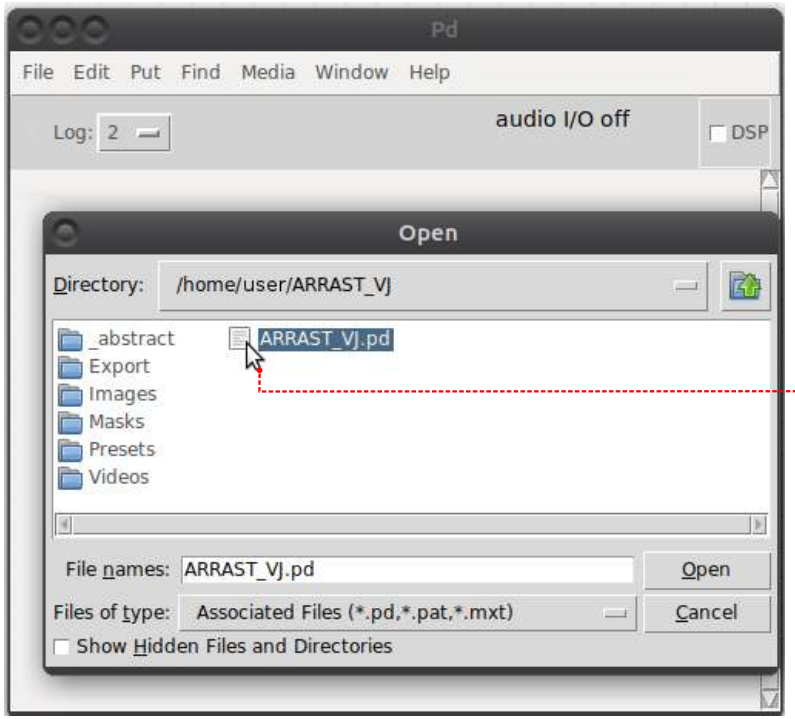
HANDS ON!

Pure Data (Pd) and GEM

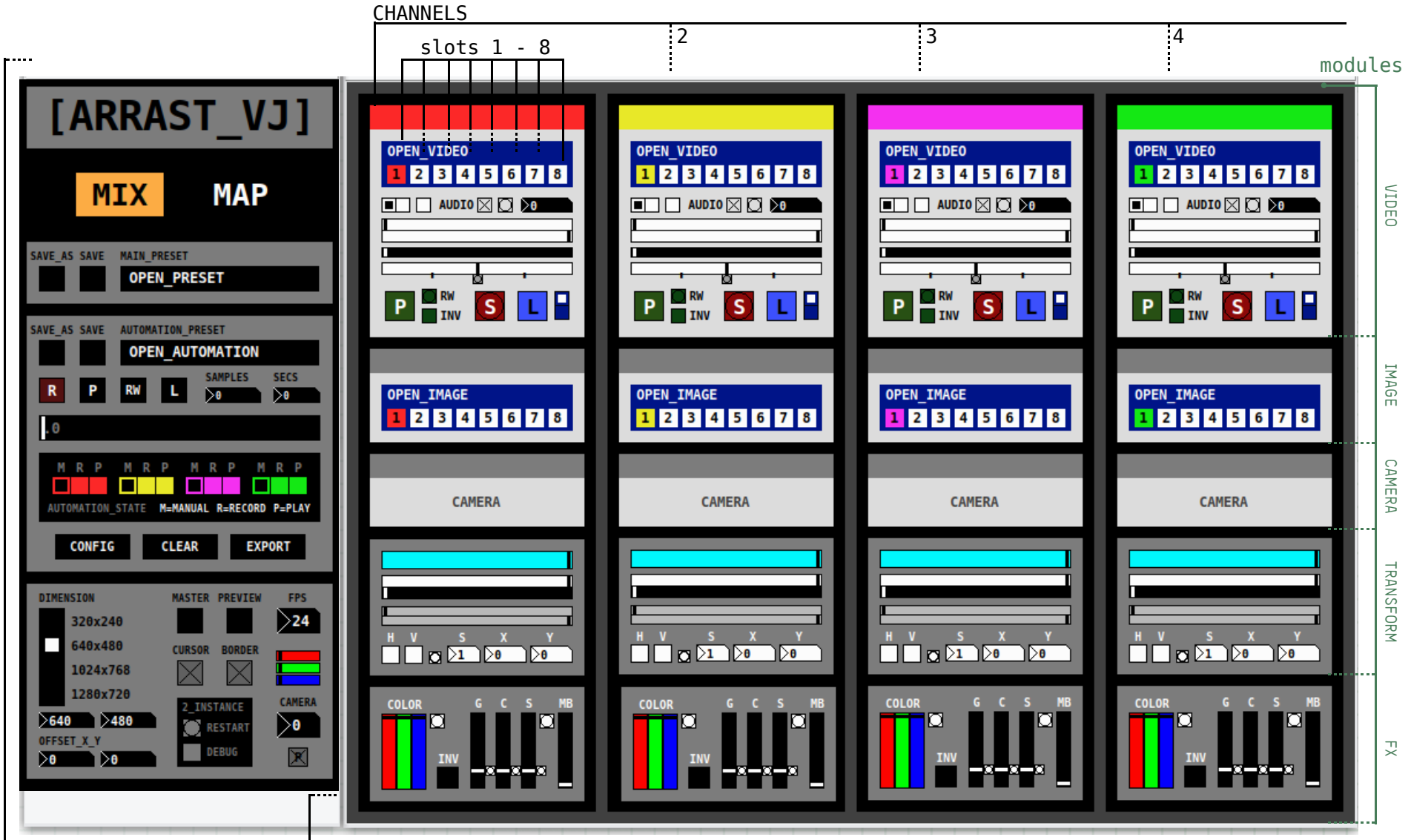
Go to your **program menu** and open **Pure Data**.

Now you are in **Pd** interface. Click on **File** > **Open** and locate **ARRAST_VJ.pd** file in the folder you've downloaded from [ARRAST_VJ]:





ARRAST_VJ.pd



[MAIN]
CONTROL WINDOW

[CHANNELS]
MIXING TOOLS WINDOW
Contains mixing modules (video / audio / image / camera / effects).

The [MAIN] control window

[ARRAST_VJ] operates in two modes: **mixing** and **mapping**.

[MIX mode] – The program opens this mode by default. These are the tools to manipulate videoclips and audio, image, camera, effects, with the four channels overlaid.

[MAP mode] – 2D surfaces mapping mode for projection – video mapping – through manual adjustment of nodes in each channel and masks application that detail the area to be projected.

You find the main coordinates to start a new project in the [MAIN] control window. To get started:

CHOOSE A RESOLUTION FOR YOUR PROJECT IN [DIMENSION]

OR TYPE CUSTOMIZED VALUES IN THE FOLLOWING FIELD.

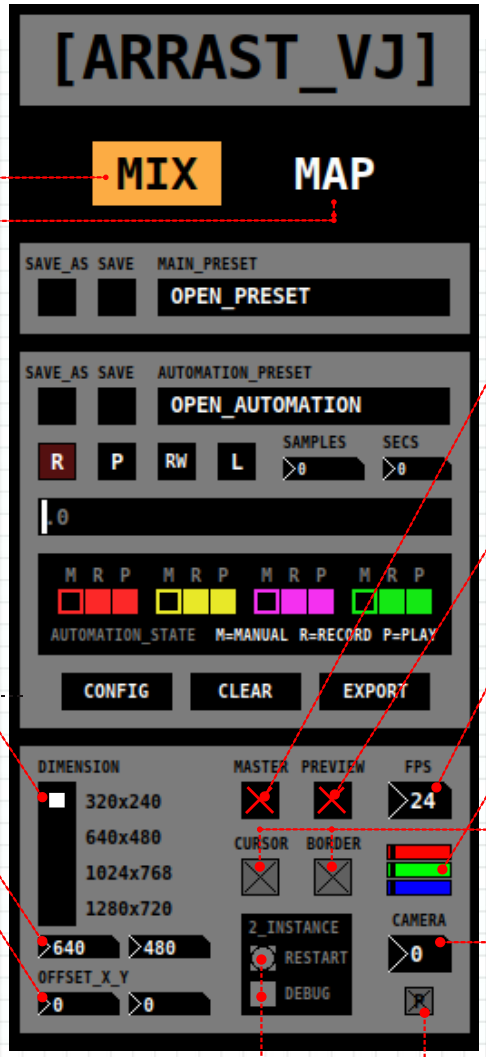
ACTIVATE THE WINDOWS:

[PREVIEW] TO VIEW THE ACTIVE CONTENT IN THE CHANNEL

AND [MASTER] TO TURN ON THE GENERAL VIDEO OUTPUT.

[MAIN]

CONTROL WINDOW



[MIX mode]

[MAP mode]

[PRESETS]

[AUTOMATION]

[DIMENSION]

general resolution of MASTER window (choose a standard on the list or insert customized values)

[OFFSET_X_Y]

Screen offset, according to horizontal (x) and vertitcal (y) axes

[2_INSTANCE]

RESTART: restarts Pure Data second instance

[DEBUG]

More information about how Pd is running

[MASTER]

Turns on and off the video output window

[PREVIEW]

Preview of the active slot content in each channel

[FPS]

Rate of video frames per second

[BACKGROUND]

Adjusts the window background colour

[CURSOR], [BORDER]

Shows cursor and shows window borders

[CAMERA]

Number of the active camera device (Linux only)

[P]

PRESET: enables memory t window configurations

[ARRAST_VJ]

[CHANNELS]

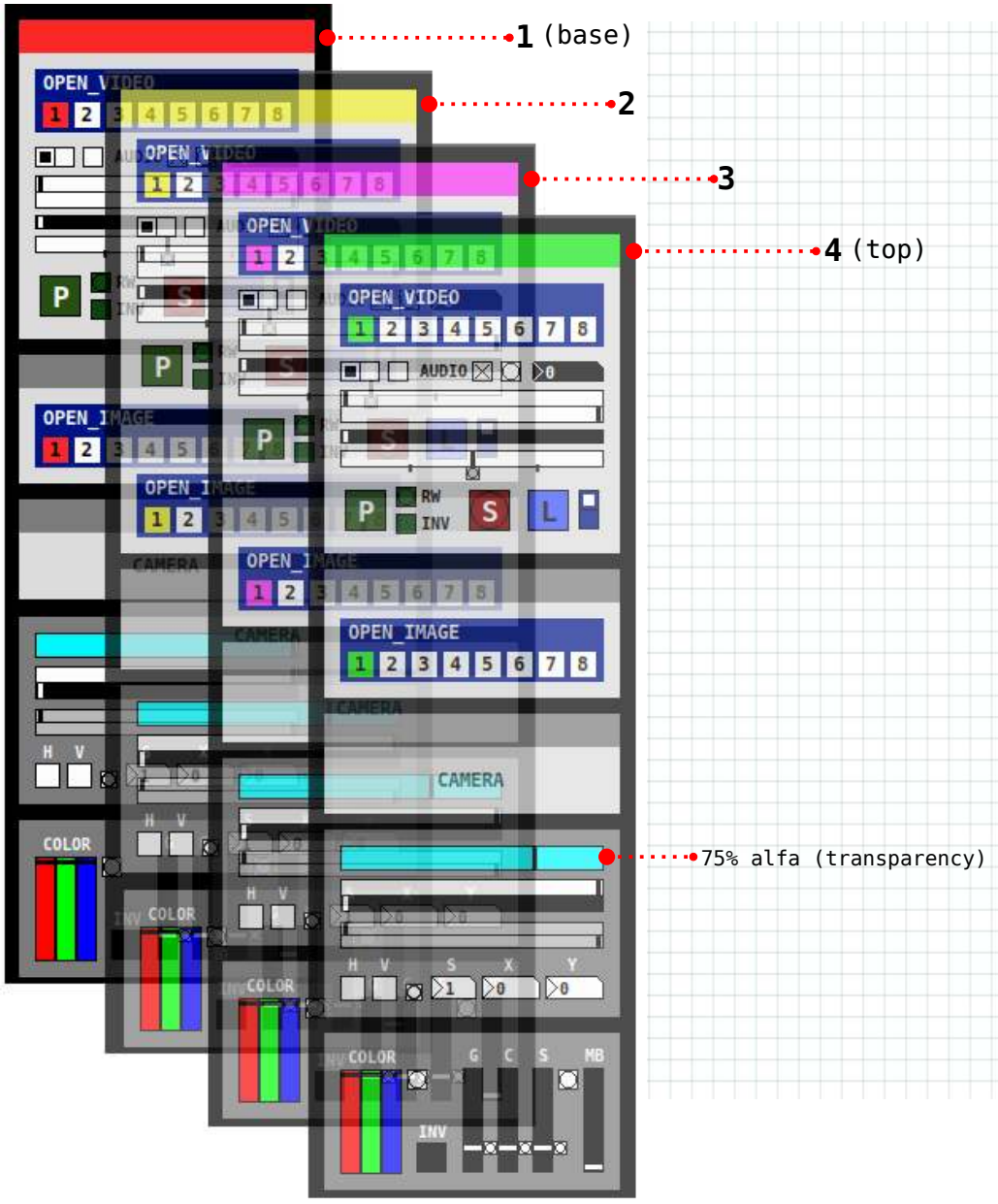
STACK OF LAYERS

The content control window is composed of four channels. Each one has three possible sources: video/audio, image and camera. Besides the sample manipulation resources, each channel has an area for transformation and for adding effects over the active material.

KEEP IN MIND: the content of channels 1 (red), 2 (yellow), 3 (magenta) and 4 (green) will be shown in **[MASTER]** window – enabled through [MAIN] window – and each one of them **acts as a layer**, always in the same stacking order.

The layer order is the same both in **MIX** and **MAP modes**.

The picture aside shows the standard organization of the channels overlaid with 75% alfa each. (25% transparency).

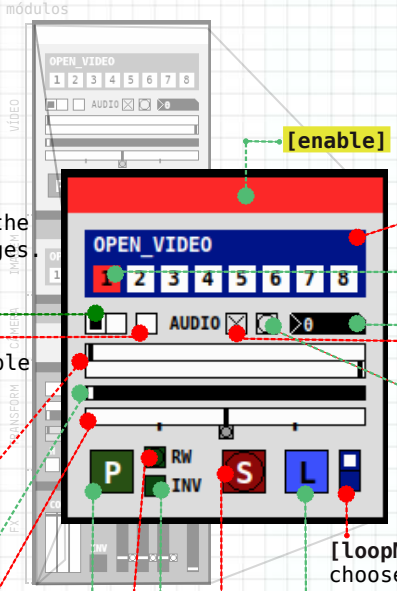


[MIX]

CHANNELS WINDOW

VIDEO

módulos



[enable] : click this colored border to **switch on and off video module.**

[IO_mode]
loop start mode:
■□ → awaits for the end of the sample to reproduce changes.
□■ → immediate reproduction.

[OPEN_VIDEO]: click to load a video (opens [ARRAST_VJ] videos folder by default).

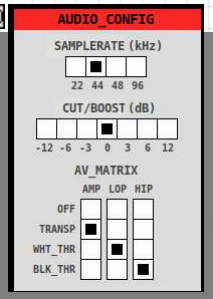
[activeSlot]: shows the active slot in the module. Each slot stores contents like video, playback data, transformation and effects added in the channel.

[IO_link]
links in and out cursors to keep the sample clipping always the same size and pastes clips from a slot to another.

[display]: sample size in seconds

[audio]: switches audio on and off (if there is a .wav file in the same folder and with the same name as the current video).

click, open →



[samplerate]: selects audio sample rate.

[boost] (CUT/BOOST): volume level selector (dB).

[AV_MATRIX]: matrix that assigns audio effects to changes in video effects. By default these settings are on (and may be changed).

[avAmp]: links volume and video transparency **[TRANSP]**.
[avLop]: links audio low pass and video white treshold **[WHT_THR]**.
[avHip]: links audio high pass and video black threshold **[BLK_THR]**.

input (in) and output (out) points on the sample selection

[inSet]: beginning of selection
[outSet]: end of selection

[sample timeline]

[play]/[pause]
enables/
disables
the sample

[loopMode]
choose between one-way and ping-pong loop modes.

[loop]
turns **on/off loop** reproduction mode

[speed]
reproduction speed
reset (in the center)

[stop]: turns off the sample

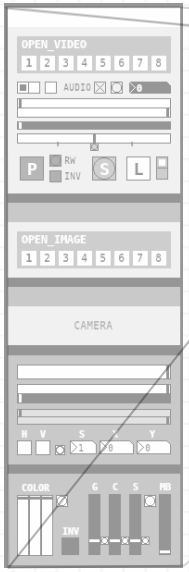
[inv]: inverts the sample

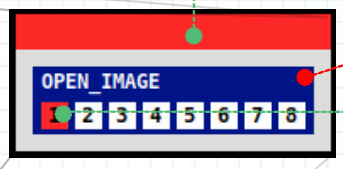
[rw]: rewinds

[MIX]

CHANNELS WINDOW

IMAGE AND CAMERA





[enable] : click this colored border to **switch on and off** the **image module**.

[OPEN_IMAGE] : choose a **.jpg** image (or .jpeg)

[activeSlot]: shows the active **slot** in the module.
each slot stores the image,t modifications and effects added in the channel.

[enable] : click this colored border to **switch on and off the camera module.**

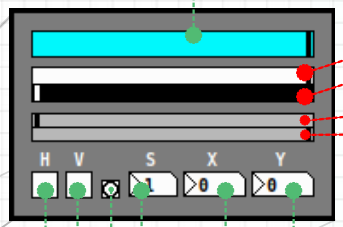
*Keep in mind that the camera can only be enabled if the [ARRAST_VJ] **[preview]** window is open.

You can locate the camera source (usb, laptop, etc) by adjusting the device number in **[camera] - [MAIN] window** (only Linux).

[MIX]

CHANNELS WINDOW

TRANSFORM



[transp] : changes **video transparency** and **audio amplitude**.

[white]: white threshold |**low pass** audio filter
[black]: black threshold | **high pass** audio filter

[fadeIn]: gradual **sample start** (until it appears)
[fadeOut]: gradual **sample end** (until it disappears)

[posY]: vertical **shift** of content

[posX]: horizontal **shift** of content

[scale]: change**size** of the image on screen

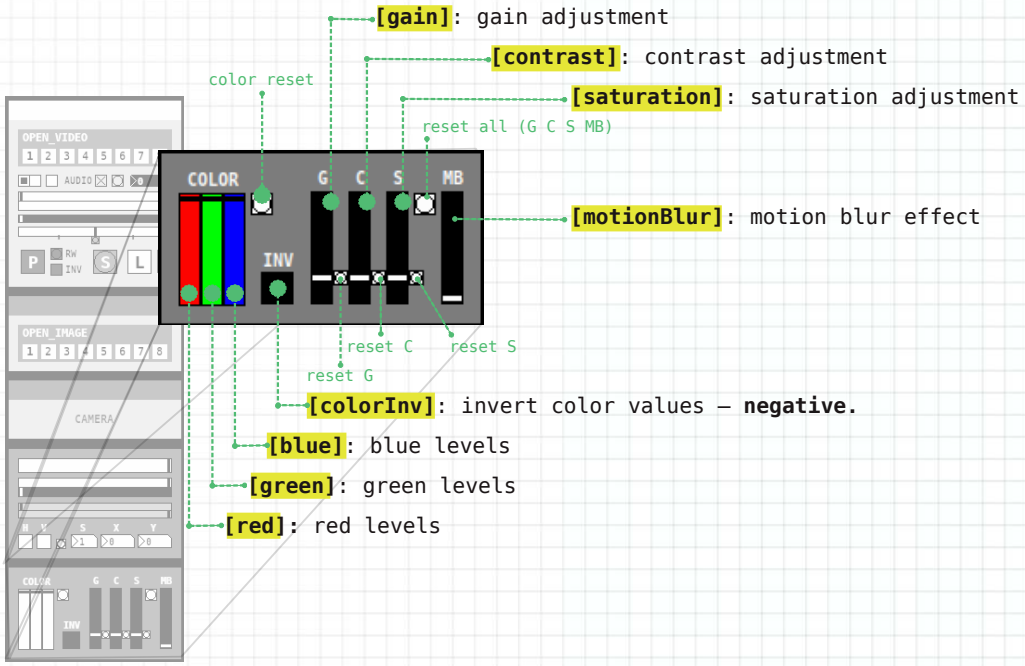
[flipV]: vertical flip

[flipH]: horizontal flip

[MIX]

CHANNELS WINDOW

FX - EFFECTS

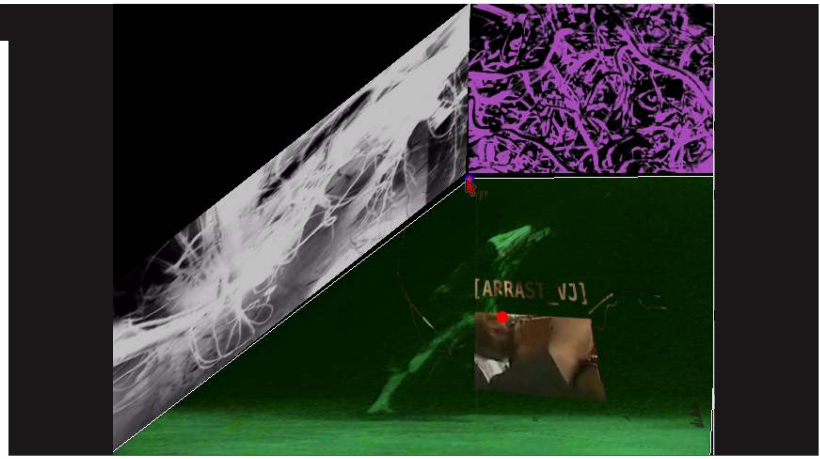


MAPPING SURFACES

[MAP] mode is accessed through [MAIN] window. Its function is to shape the four video outputs into surfaces and into non-conventional projector types, allowing the mixed content **not only to be used but also to be adapted to the environment**, with techniques that are nowadays called **videomapping**.

So the mapped **shape defines how the active content in each channel** is going to be projected. It can be manually adjusted or preset as a mask (.jpg format).

You can check how the [MAP] mode video output is displayed via **[MASTER]** window: at first channels are set side by side but you can change the display in node adjustment or by clicking **[FULL]**. **[RESET]** returns channels to their initial position.



JANELA **[MASTER]**

MANUAl ADJUSTMENT

Click **[MAP]** on **[MAIN]** window .

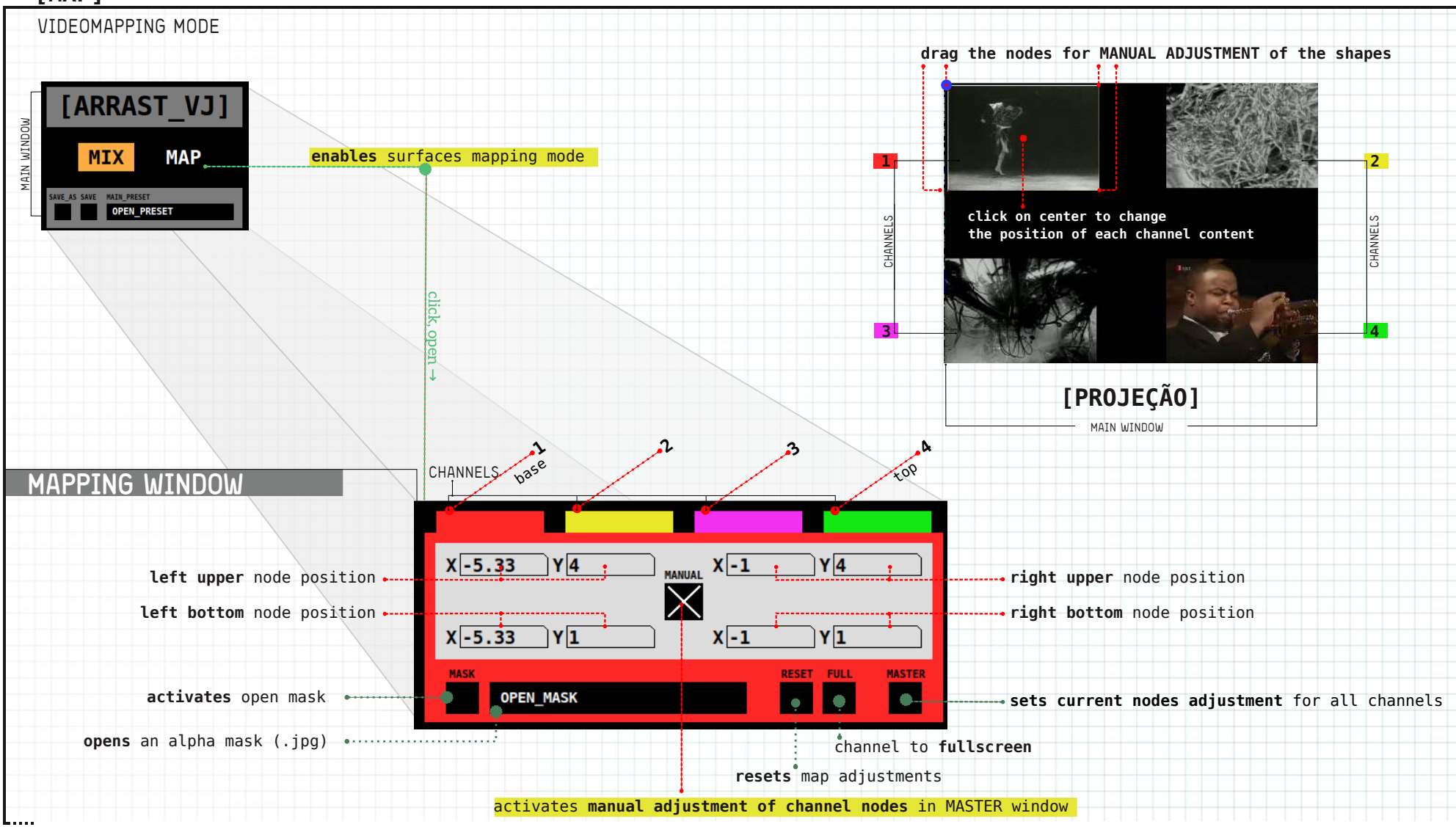
When **[MAP]** window opens, four tabs are shown, one for each channel - signaled by color.

Activate manual edition of channel mapping by clicking the  button in the center of the window.

Now pay attention at **[MASTER]**, **the video output**: there you can change shape and position of the channel according to the projection screen:
→ **Shape**: drag the channel nodes (the red dots around the corners) until you get the desired shape.
→ **Position**: click the central circle t**to move it.**

[MAP]

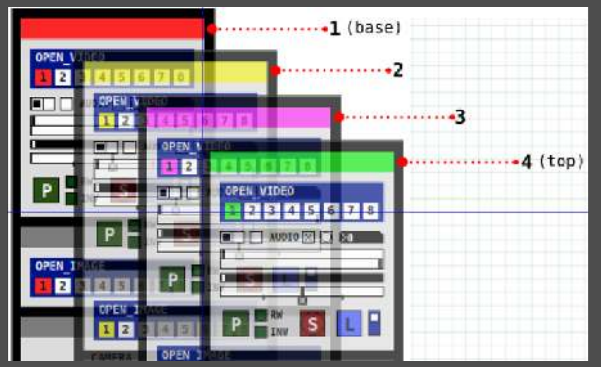
VIDEOMAPPING MODE



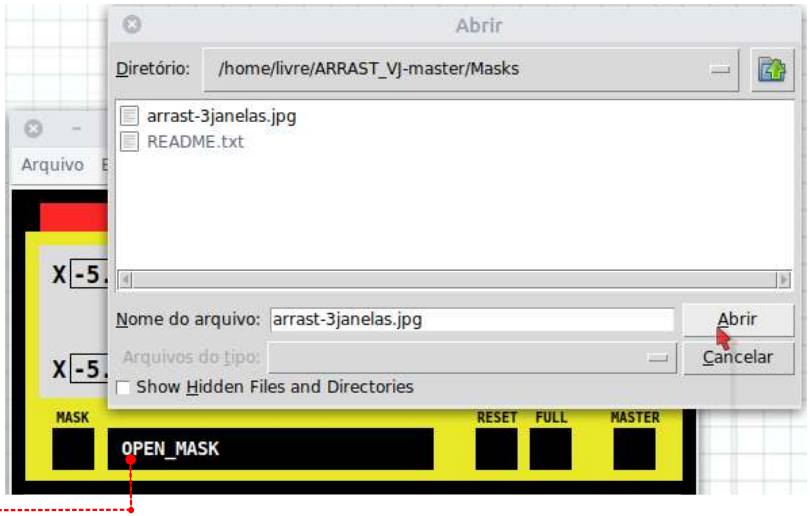
APPLYING MASKS

Prepare an image file in black and white for it to operate as a mask*. The black areas will be hidden and white areas will work like windows to be filled by the video. Gray areas have intermediate masking effect, therefore they allow equivalent luminosity to pass through. This is a resource for experimenting with different transparency levels and transition gradients.

Save the mask in .jpg format with the same resolution of the current project. Go to **[MAPPING]** window, click **[OPEN_MASK] to select file** and then **"open"**. Click then on **[MASK]** to **enable** it .



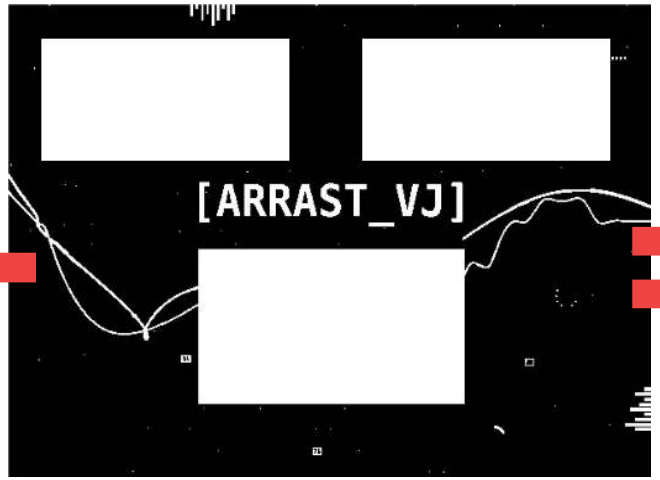
IMPORTANT: MASKS FOLLOW THE SAME ORDER OF THE CHANNEL LAYERS STACK.



* A mask added to a channel follows the quadrilateral shape that supports this channel projection.

In the example below a black and white mask was used, with no intermediate shades of gray. For the black covered area it is like the projector LEDs could not pass through the mask, and for the white areas, luminosity permeates 100%.

ACTIVE CONTENT ON CHANNEL



ADDED AND ACTIVE MASK

RESULT

MASK APPLICATIONS THROUGH
SURFACE VIDEOMAPPING TECHNIQUE

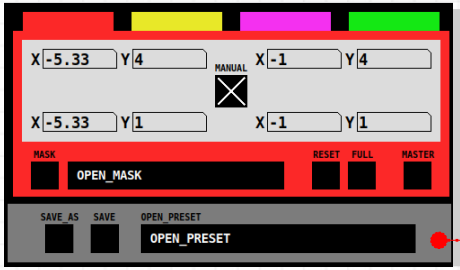
[RECORD, AUTOMATE, EXPORT]

MAIN WINDOW

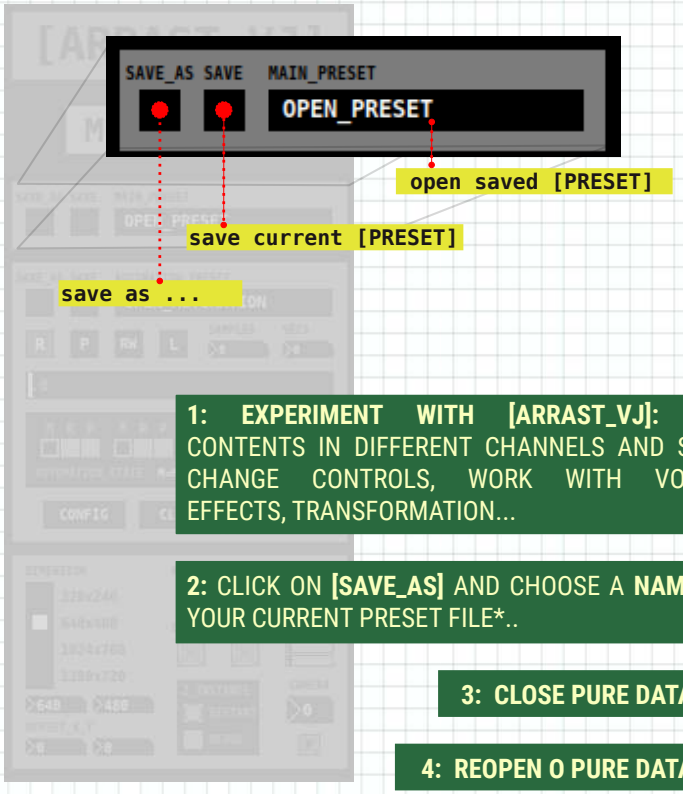
PRESETS

When operating [ARRAST_VJ] you can **record** all parameters **the state they are in, as presets files.**

These files set the state of each [MIX mode] resource in the moment they are generated: **main window** and **control modules** of video, image, camera, transformation and effects. **When reopening one of these files**, [PRESET] rearranges the current window, recreates what was going on and reopens the channels content.



ATTENTION: THERE IS AN EXCLUSIVE PRESETS SYSTEM IN [MAP MODE] TO SAVE PROJECTION SHAPES AND MASKS. THIS MENU IS IN THE BOTTOM OF [MAPPING WINDOW].



1: EXPERIMENT WITH [ARRAST_VJ]: LOAD CONTENTS IN DIFFERENT CHANNELS AND SLOTS, CHANGE CONTROLS, WORK WITH VOLUME, EFFECTS, TRANSFORMATION...

2: CLICK ON **[SAVE_AS]** AND CHOOSE A **NAME** FOR YOUR CURRENT PRESET FILE*..

3: CLOSE PURE DATA (Pd).

4: REOPEN O PURE DATA (Pd).

5: INSIDEPd, **REOPEN ARRAST_VJ.pd**

6: CLICK ON [OPEN_PRESET] AND CHOOSE THE PRESET YOU JUST SAVED.

* Mapping [PRESETS] are saved in /Presets/Map.

[RECORD, AUTOMATE, EXPORT]

MAIN WINDOW

AUTOMATION

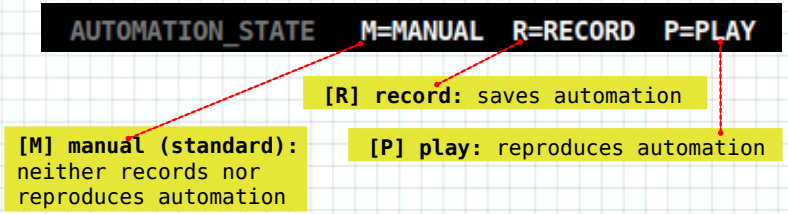
The concept of **[AUTOMATION]** is an [ARRAST_VJ] property that allows **recording in a timeline of virtually all parameters manipulated in the program**. This feature stores a sequence of all sliders and mixing interface buttons for a particular time, for example.

It is possible then to operate new gestures over very complex and cumulative loops, which also allows automation to be overwritten shifting between writing and reading modes of the timeline.

This "save as automation" scheme optimizes the programming of performative gestures and enables indexing movements as numerical tags (like text files), producing outputs that are apart from audiovisual file renderings.

The same input movement of a composition process, whether effects or transitions, may be applied in different movie layers arrangements, creating gesture patterns as editable and consistent groups for compositional language.

IMPORTANT: THESE ARE THE POSSIBLE AUTOMATION STATES FOR EACH RECORDED PARAMETER:



* By default we recommend to save automations inside ARRAST_VJ file structure: /Presets/Automation.

[RECORD, AUTOMATE, EXPORT]

MAIN WINDOW

AUTOMATION - INTERFACE

save as ...

save current [AUTOMATION]

open saved [AUTOMATION]

TIMELINE [CONTROLS]

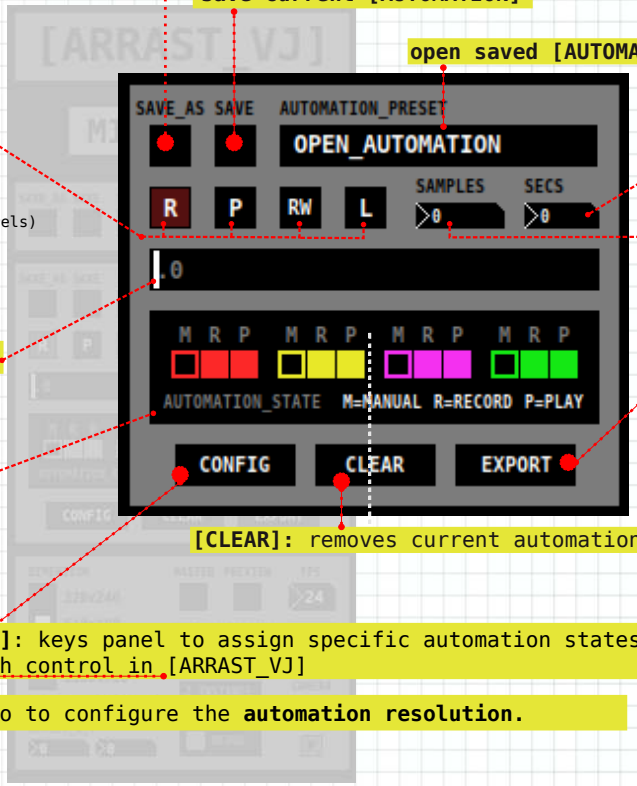
- **rec:** sets recording
- **play:** reproduces timeline and starts the recording time
- **rewind:** restarts timeline reproduction (also restarts all active samples in channels)
- **loop:** plays timeline in loop when enabled

automation [TIMELINE]

Click play [P] to activate

automation [KEYS] by channel

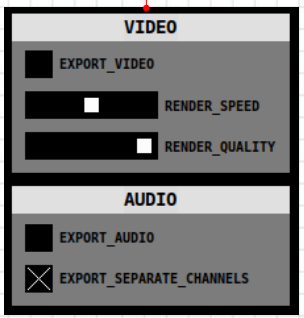
- **[M] manual:** doesn't record
- **[R] record:** records automation
- **[P] play:** reproduces automation



[SECS]: timeline count in seconds, it allows manual input of a specific period

[SAMPLES]: stored data count

[EXPORT]: opens a dialog box to export current [AUTOMATION] timeline to audio (.wav) and video (frames sequence*)



[CLEAR]: removes current automation record

[CONFIG]: keys panel to assign specific automation states for each control in [ARRAST_VJ]

and also to configure the **automation resolution.**

[RECORD, AUTOMATE, EXPORT]

MAIN WINDOW

AUTOMATION - STEP BY STEP

1: START FROM A SAVED AND OPEN PRESET (CHECK PREVIOUS CHAPTER).

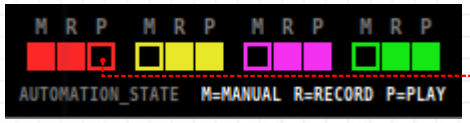
2: TO AUTOMATE ONLY THE FIRST CHANNEL: IN AUTOMATION KEYS, **ASSIGN RECORD MODE** FOR CHANNEL 1 (RED):

3: CLICK ON R **TO SET RECORD UP AND** R **TO PLAY TIMELINE AND RECORD.**

4: EXPERIMENT ON THE CHANNEL THAT IS ON RECORD MODE, CHANGING EVERY KIND OF CONTROL. TRY ALSO IN OTHER CHANNELS.

5: STOP RECORDING AND REPRODUCTION BY DISABLING R AND P.

6: REWIND THE TIMELINE WITH RW .



7: ON KEYS, ASSIGN P MODE (PLAY) FOR CHANNEL 1 (RED).

8: AND THEN JUST ACTIVATE R **TO REPRODUCE AUTOMATION.**

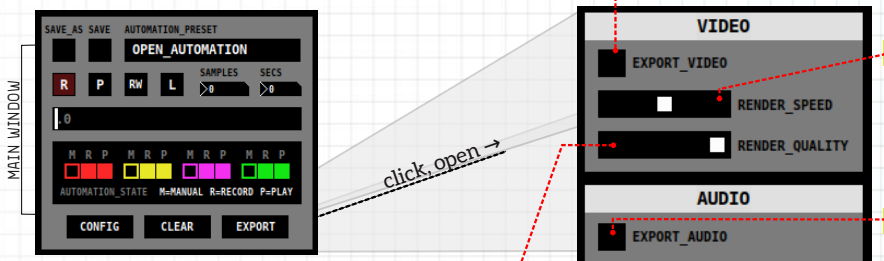
9: GO TO AUTOMATION MODULE ON MAIN WINDOW. **CLICK [SAVE_AS]** AND **CHOOSE A NAME AND FOLDER* TO SAVE YOUR AUTOMATION.**

* By default we recommend to save automations inside ARRAST_VJ file structure: /Presets/Automation /Presets/Automation.

[RECORD, AUTOMATE, EXPORT]

MAIN WINDOW

EXPORT



[EXPORT_VIDEO]: renders automation frames as .jpg files in export folder

[RENDER SPEED]:rendering speed in frames per second (1 / 4/8/12/16 FPS) – the bigger the video resolution and the more effects applied, the LOWER must be RENDER_SPEED.

[EXPORT_AUDIO]: exports automation audio as audio_mix.wax file

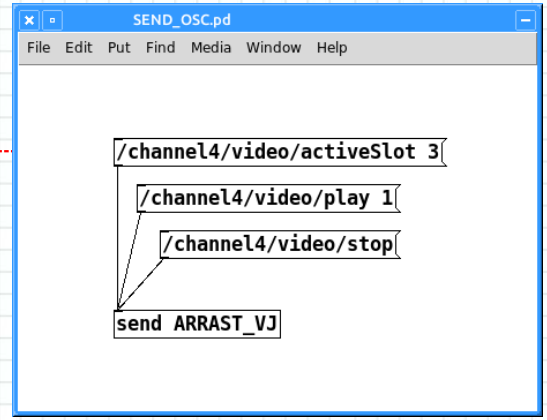
[EXPORT_SEPARATE_CHANNELS]: if enabled, exports audio files split by channel, besides the mix with all channels

[RENDER_QUALITY]
quality of the .jpg images created in the export folder

EXTENSIONS

Apart from the mixing and videomapping interface presented, [ARRAST_VJ] allows developing interactive extensions for user experience. All the controls here described are associated with OSC communication protocol and may serve as a door to connect the program to external controllers and customized graphic interfaces.

Via OSC, it is possible to externally control any [ARRAST_VJ] parameter. Following patterns described in OSC_TREE file (in [ARRAST_VJ] folder), just send the messages by using a Pd patch with an object [send ARRAST_VJ] or through network at 7777 computer port.



CREDITS

This user guide was created with **Inkscape, Gimp and Kazam free softwares**.

Simone Bittencourt Azevedo
Editorial Design, Writing, Layout

Elisa Tkatschuk / Simone Bittencourt Azevedo
English Translation & Review

Priscila Rodrigues
Photography

Rafael De Marchi
ARRAST_VJ Visual Identity & Cover

Thayná Mallmann / Bruno Rohde
Technical Advice, Final Review

CONTACT

Comments and questions can be sent to **contato@arrastvj.org**.

FINANCIAL SUPPORT

Fundo de cultura

SECRETARIA DA FAZENDA

SECRETARIA DE CULTURA

BAHIA GOVERNO DO ESTADO