

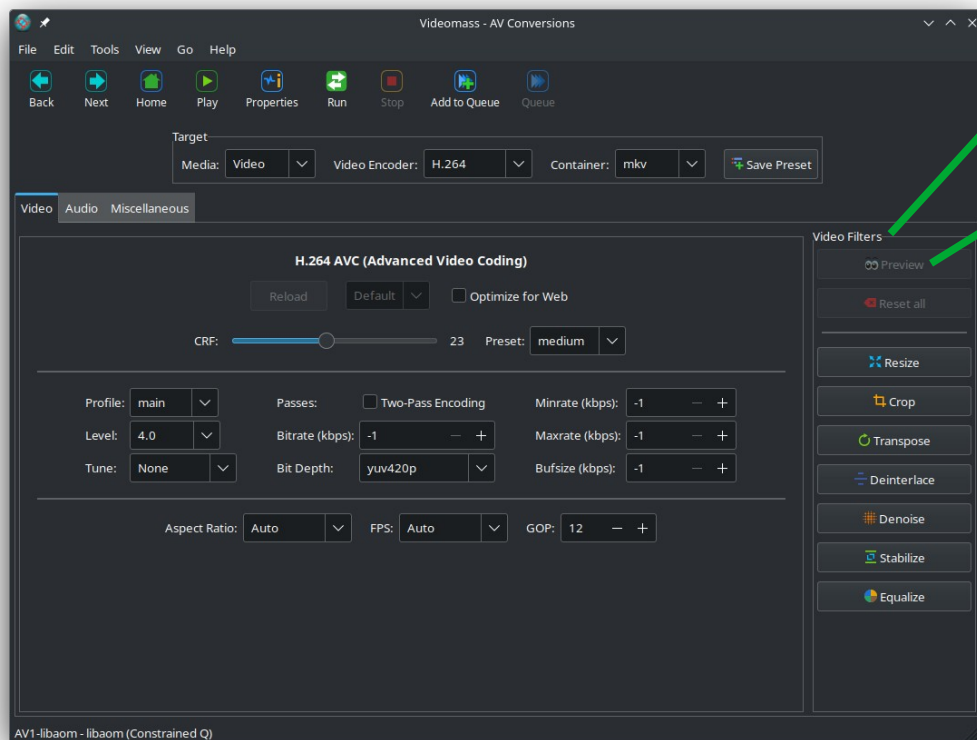
# Videomass

## Video filters on AV-Conversions

The AV-Conversions function features video filters equipped with sophisticated graphic interfaces. The default order in which the video filter chain is applied is as follows:

*deinterlace/interlace* → *denoise* → *stabilize* → *unsharp* → *crop* → *resize* → *setdar* → *setsar* → *transpose* → *eq*

To save a profile on the Presets Manager including the filter chain settings, simply click the Save Preset button. See the following video example: [https://youtu.be/s92H36\\_yBXw](https://youtu.be/s92H36_yBXw)



Video filters available with GUI.

Preview has the ability to chain some enabled video filters.

# Index

- [Resize \(scale filter\)](#)
- [Crop \(crop filter\)](#)
- [Transpose \(transpose filter\)](#)
- [Deinterlace \(w3fdif/yadif/interlace filters\)](#)
- [Denoise \(nlmeans/hqdn3d filters\)](#)
- [Stabilize\(vidstabdetect/vidstabtransform/unsharp filters\)](#)
- [Equalize \(eq filter\)](#)

# Resize (scale filter)

Resize the input video or image to change the resolution result.

## Documentation

<https://ffmpeg.org/ffmpeg-filters.html#scale>

<https://trac.ffmpeg.org/wiki/Scaling>

## Box New size in pixel

If you want to keep the aspect ratio, you have to select the checkbox and specify only one component (width or height), and set the other component (width or height) to -1 or -2. The **setdar** and **setsar** functions will be disabled.

Any value of 0 means to disable a component and implies the cancellation of the setting.

**Width:** sets the width in pixels of the output video.

**Height:** sets the height in pixels of the output video.

## Box Aspect Ratio (setdar and setsar)

Any value at 0 means to disable a component and implies the cancellation of the setting.

**Setdar:** set the frame (d)isplay (a)spect (r)atio. The setdar filter sets the display aspect ratio for the output video. The result to be entered corresponds to the fractional unit of the numerator and denominator, for example: 16/9.

**Setsar:** (S)ample (known as Pixel) (A)spect (R)atio for the output video. As with setdar, the result to be entered corresponds to the fractional unit of the numerator and denominator, for example: 1/1.

## Examples

To reduce a video in 1280X720 to 640X360, then keeping the same aspect ratio to 16/9 there are 3 ways:

- 1) set width to 640 pixel and set height to 360
- 2) set width to 640 pixel and set height to -1 or -2
- 3) set height to 360 pixel and set width to -1 or -2

setdar/setsar (aspect ratio) should not need to be set.

To reduce the same video to a resolution of 640X480, we should also set the 4/3 setdar filter.

To change the resolution on undefined aspect ratio (400X200), you can use setsar filter: setsar to 1:1

Confirm your choices with the **OK** button. The **Reset** button, restores the values to defaults parameters and disables the filter if you confirm with the OK button. However, the values equal to 0 set on all input fields disable this filter.

# Crop (crop filter)

The crop filter crops specific portions in the frame area.

## Documentation

<https://ffmpeg.org/ffmpeg-filters.html#crop>

## Cropping videos is now very easy

Just place the mouse cursor in the area of the frame displayed in the preview and select the interested area. You should now see a colored clipping mask. Once satisfied, confirm the setting with the OK button.

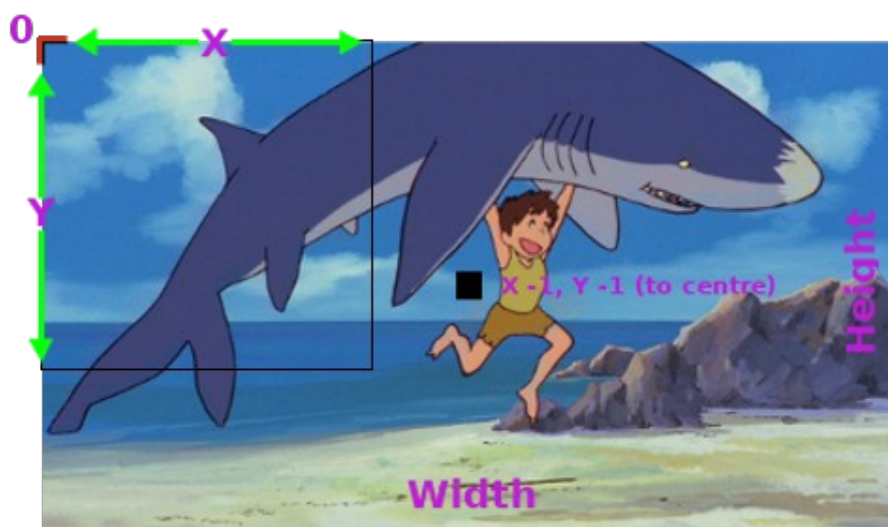
## Technical details

The crop filter is used by specifying values in pixels on the coordinates concerning height, width, horizontal distance (Y) and vertical distance (X). The starting point of the X and Y coordinates always starts from 0, which is the upper left end of the frame (the highlighted red corner of the image below). Each of the coordinates can only be disabled with the value -1 independently, which implies the centering of the frame (see image below).

**Height** and **Width** represent the area selected for cropping.

**X**, The horizontal position of the left edge. The value 0 sets the position on the extreme left of the frame. Values above 0 move the position to the right side of the frame. The -1 value to center the frame horizontally.

**Y**, The vertical position of the top edge of the left corner. Values above 0 move the position towards the bottom side of the frame. The -1 value to center the frame vertically.



Confirm your choices with the **Ok** button. The **Reset** button restores the default values and will disable the filter if re-confirmed. Values equal to 0 set to width and height disable this filter.

## Transpose (transpose filter)

The transpose filter allows you to rotate the video image at certain degrees.

### Documentation

<https://ffmpeg.org/ffmpeg-filters.html#transpose>

### Rotation settings

The frame can be rotated right, left or upside down. The Reset button will restore the original position

Confirm your choices with the **OK** button. The **Reset** button restores the default values and will disable the filter if re-confirmed.

## Deinterlace (w3fdif/yadif/interlace filters)

Videomass offers an interface with two different types of video de-interlacers, and a simple interlacer with the related options.

Deinterlacers are useful for removing a series of annoying little lines that disturb the image of a movie. Although not very useful, an interlacer for progressive content is also available.

### Documentation

#### w3fdif:

<http://underpop.online.fr/f/ffmpeg/help/w3fdif.htm.gz>

<https://ffmpeg.org/ffmpeg-filters.html#w3fdif>

#### yadif:

<https://ffmpeg.org/ffmpeg-filters.html#yadif-1>

#### interlace:

<https://ffmpeg.org/ffmpeg-filters.html#interlace>

## Deinterlacing

- **w3fdif** stands for *Weston 3 Field Deinterlacing Filter*. Based on the process described by Martin Weston for BBC R&D, and implemented based on the de-interlace algorithm written by Jim Easterbrook for BBC R&D, the Weston 3 field deinterlacing filter uses filter coefficients calculated by BBC R&D.

### Advanced options:

**filter**, set the interlacing filter coefficients. Accepts one of the following values:

- *simple* - Simple filter coefficient set.
- *complex* - More-complex filter coefficient set (default)

**deint**, specify which frames to deinterlace. Accept one of the following values:

- *all* - Deinterlace all frames (default)
- *interlaced* - Only deinterlace frames marked as interlaced.

- **yadif** deinterlace the input video

yadif means (y)et (a)nother (d)e (i)nterlacing (f)ilter.

### Advanced options:

**Mode**, the interlacing mode to adopt. It accepts one of the following values:

- *0, send\_frame* - Output one frame for each frame.
- *1, send\_field* - Output one frame for each field. (default)
- *2, send\_frame\_nospatial* - Like *send\_frame*, but it skips the spatial interlacing check.
- *3, send\_field\_nospatial* - Like *send\_field*, but it skips the spatial interlacing check.

**Parity**, the picture field parity assumed for the input interlaced video. It accepts one of the following values:

- *0, tff* - Assume the top field is first.
- *1, bff* - Assume the bottom field is first.
- *-1, auto* - Enable automatic detection of field parity (default). If the interlacing is unknown or the decoder does not export this information, top field first will be assumed.

**Deint**, specify which frames to deinterlace. Accept one of the following values:

- *all* - Deinterlace all frames (default)
- *interlaced* - Only deinterlace frames marked as interlaced.

## Interlacing

- **interlace** simple interlacing filter from progressive contents. This interleaves upper (or lower) lines from odd frames with lower (or upper) lines from even frames, halving the frame rate and preserving image height.

Advanced options:

**scan**, determines whether the interlaced frame is taken from the even (tff - default) or odd (bff) lines of the progressive frame.

**low pass**, enable (default) or disable the vertical lowpass filter to avoid twitter interlacing and reduce moire patterns. Default is no setting.

## Denoise (nlmeans/hqdn3d filters)

Videomass supports two of the most popular denoisers used by FFmpeg: **nlmeans** and **hqdn3d**

### Documentation

#### **nlmeans**

<https://ffmpeg.org/ffmpeg-filters.html#nlmeans>

#### **hqdn3d**

<https://ffmpeg.org/ffmpeg-filters.html#hqdn3d>

### Why nlmeans and hqdn3d ?

The reason for this choice is that the denoiser **nlmeans** exists only on newer versions of FFmpeg, while **hqdn3d** exists on both new and older versions of FFmpeg. When one of them fails, try the other.

**nlmeans** denoise frames using Non-Local Means algorithm is capable of restoring video sequences with even strong noise. It is ideal for enhancing the quality of old VHS tapes.

**hqdn3d** this is a high precision/quality 3d denoise filter. It aims to reduce image noise, producing smooth images and making still images really still. It should enhance compressibility.

# Stabilize (vidstabdetect / vidstabtransform / unsharp)

The GUI for the Video Stabilizer filter was introduced on Videomass from version 3.4.0

## Why is it needed

A video acquired using a hand-held camera or a camera mounted on a vehicle, typically suffers from undesirable shakes and jitters. Activities such as surfing, skiing, riding, bike and walking while shooting videos are especially prone to erratic camera shakes. Vidstab targets these video contents to help create smoother and stable videos.

## Documentation

- <http://www.ffmpeg.org/ffmpeg-filters.html#toc-vidstabdetect-1>
- <http://www.ffmpeg.org/ffmpeg-filters.html#toc-vidstabtransform-1>
- Home page: <http://public.hronopik.de/vid.stab/?lang=en>
- Development page: <https://github.com/georgmartius/vid.stab>

## Descriptions

Vidstab uses an algorithm capable of making shaky footage really stable with the default settings.

At the time of writing, it has not yet been included by default in FFmpeg, requiring specific compilation to enable **libvidstab** support. However, some FFmpeg releases are available for Linux, Windows and MacOs statically compiled with libvidstab support.

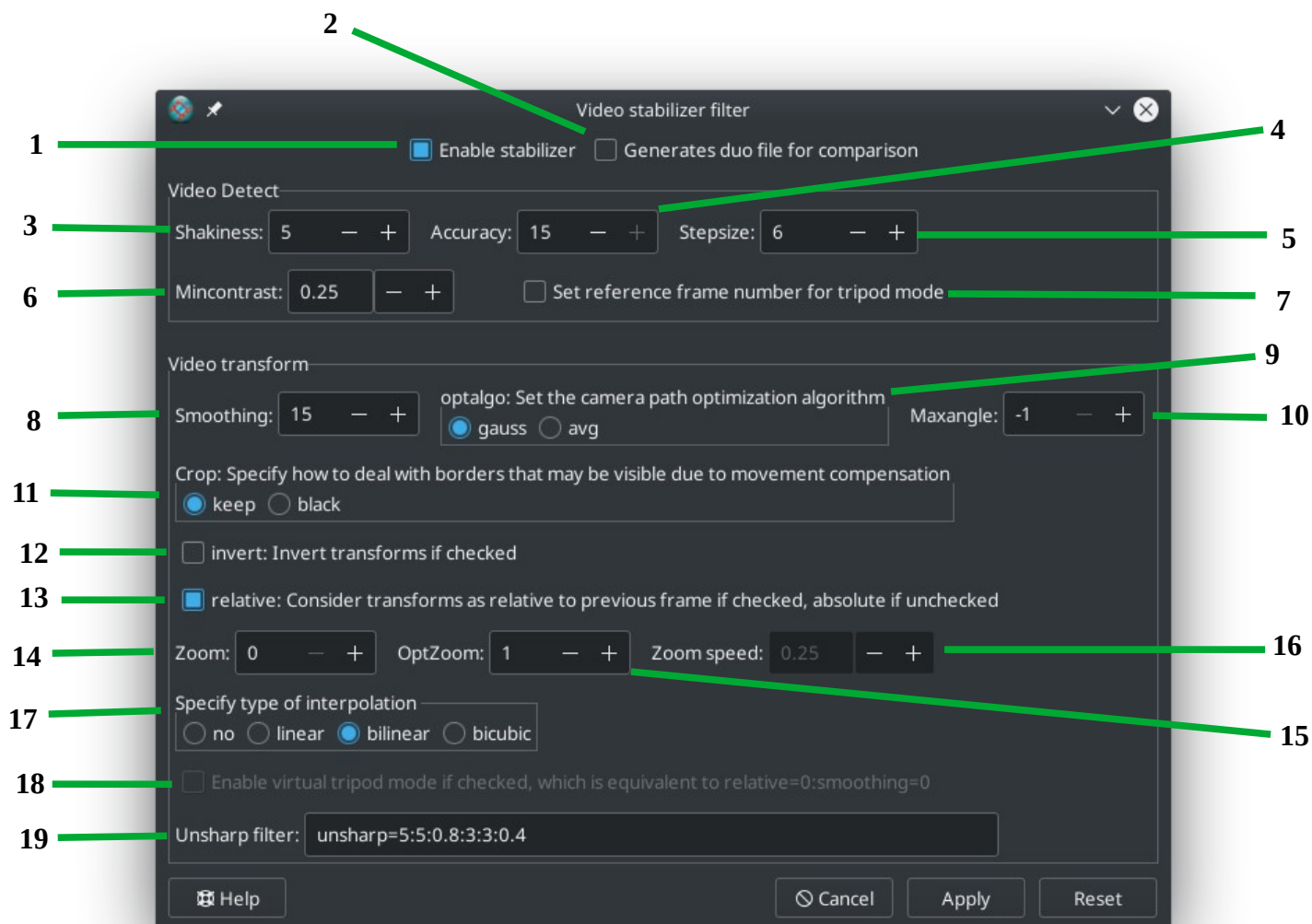
## Features included

- Fast detection of subsequent transformations e.g. translation and rotations up to a given extent.
- Low pass filtered smoothing with adjustable horizon.

### Detection algorithms:

- Smart and fast multi measurement fields algorithm with contrast selection.
- Brute force algorithm only for translations.
- Clipping options: keep blank (black) or keep from previous frames.
- Optional drawing of measurement fields and detected transformations for visual analysis.
- Zooming possible to get rid of jiggling borders (automatic mode).
- Resulting images are interpolated (different algorithms).
- Sharpening of the stabilized movie to compensate for interpolation effects due to rotation/zooming (only with Transcode).
- Single pass filter for streaming applications(only with Transcode).
- Virtual-tripod-mode to get a tripod experience.





1. The check-box to enable or disable this filter (confirming with the OK button)
2. If enabled, this exclusive Videomass feature allows you to preview a duo file by means of an additional task. A duo file is a dual-view video, with the original video on the left and the processed video on the right, for the purpose of evaluating the result.
3. **Shakiness:** Set the shakiness of input video or quickness of camera. It accepts an integer in the range 1-10, a value of 1 means little shakiness, a value of 10 means strong shakiness. Default value is 5.
4. **Accuracy:** Set the accuracy of the detection process. It must be a value in the range 1-15. A value of 1 means low accuracy, a value of 15 means high accuracy. Default value is 15.
5. **Stepsize:** Set stepsize of the search process. The region around minimum is scanned with 1 pixel resolution. Default value is 6.
6. **Mincontrast:** Set minimum contrast. Any measurement field having contrast below this value is discarded. Must be a floating point value in the range 0-1. Default value is 0.3.

7. **Tripod**: Set reference frame number for tripod mode. If enabled, the motion of the frames is compared to a reference frame in the filtered stream, identified by the specified number. The intention is to compensate all movements in a more-or-less static scene and keep the camera view absolutely still. Default is disabled, the frames are counted starting from 1.  
NOTE: If this mode is used in first pass then it should also be used in second pass.
8. **Smoothing**: Set the number of frames ( $\text{value} * 2 + 1$ ), used for lowpass filtering the camera movements. Default value is 15.  
For example, a number of 10 means that 21 frames are used (10 in the past and 10 in the future) to smoothen the motion in the video. A larger value leads to a smoother video, but limits the acceleration of the camera (pan/tilt movements). 0 is a special case where a static camera is simulated.
9. **Optalgo**: Set the camera path optimization algorithm. Accepted values are:  
**gauss**: Gaussian kernel low-pass filter on camera motion (default).  
**avg**: Averaging on transformations.
10. **Maxangle**: Set maximal angle in radians ( $\text{degree} * \pi / 180$ ) to rotate frames. Default value is -1, meaning: no limit.
11. **Crop**: Specify how to deal with empty frame borders that may be shrinked-in due to movement compensation. Available values are:  
**keep**: Keep image information from previous frame (default).  
**black**: Fill the border-areas black.
12. **Invert**: Inverti trasformazioni se abilitato. Il valore predefinito è disabilitato. Invert transforms if checked. Default value is unchecked.
13. **Relative**: Consider transforms as relative to previous frame if checked, absolute if unchecked. Default value is unchecked (disabled).
14. **Zoom**: Set percentage to zoom. A positive value will result in a zoom-in effect, a negative value in a zoom-out effect. Default value is 0 (no zoom).
15. **Optzoom**: Set optimal zooming to avoid blank-borders. Accepted values are:  
**0**: Disabled.  
**1**: Optimal static zoom value is determined (only very strong movements will lead to visible borders) (default).  
**2**: Optimal adaptive zoom value is determined (no borders will be visible), see **zoomspeed**. Note that the value given at zoom is added to the one calculated here.
16. **Zoomspeed**: Set percent to zoom maximally each frame (enabled when optzoom is set to 2). Range is from 0 to 5, default value is 0.25.
17. **Interpol**: Specify type of interpolation. Available values are:  
**no**: No interpolation.  
**linear**: Linear only horizontal.  
**bilinear**: Linear in both directions (default).  
**bicubic**: Cubic in both directions (slow speed).

18. **Tripod**: Abilita la modalità treppiede virtuale se abilitata, che è equivalente a `relative=0`: `smoothing=0`. Il valore predefinito è disabilitato. NOTA: se questa modalità è stata utilizzata nel primo passaggio, dovrebbe essere usata anche nel secondo passaggio. Enables virtual tripod mode if enabled, which is equivalent to **relative=disabled** and **smoothing=0**. Default is unchecked (disabled).

NOTE: If this mode has been used in first pass then only it should be used in second pass.

19. **Unsharp**: This filter should always be used for best results.

## Equalize (eq)

The equalize filter set brightness, contrast, saturation and approximate gamma adjustment.

### Documentation

- <https://ffmpeg.org/ffmpeg-filters.html#eq>

### Description

The graphical interface for this video filter was created to allow easy and immediate color management. It features a dual preview display useful for comparing effects against the source frame.

Included is the ability to search for specific time points in the movie by moving the frame search slider.

Confirm your choices with the **OK** button. The **Reset** button restores the default values and will disable the filter if re-confirmed.