

XviD Options Explained



This is the "XviD Options Explained" for the XviD codec on Windows.

Maintained by Koepi (de_koepi@lycos.de).

Versions/Changes:

- 1.0 17.02.2002 Initial release of a first draft.
- 1.0 28.02.2002 Plenty of corrections and more precise explanations
- 1.1 02.03.2002 A lot of cleanups thanks to Jon Ingram :)
- 1.2 04.03.2002 Added quantization matrix, removed quant. smoothing
- 1.3 08.03.2002 Explanations for Nic's XviD Directshow Filter, cleanups, restructured the document

XviD's history

XviD is an MPEG4 video codec which is being developed under the open source model, much like the LAME MP3 encoder. It's based upon the efforts made in the Open Source project "OpenDivX" by DivX Networks, which shut it down and closed the source with all the ideas and work spent by many coders from the net.

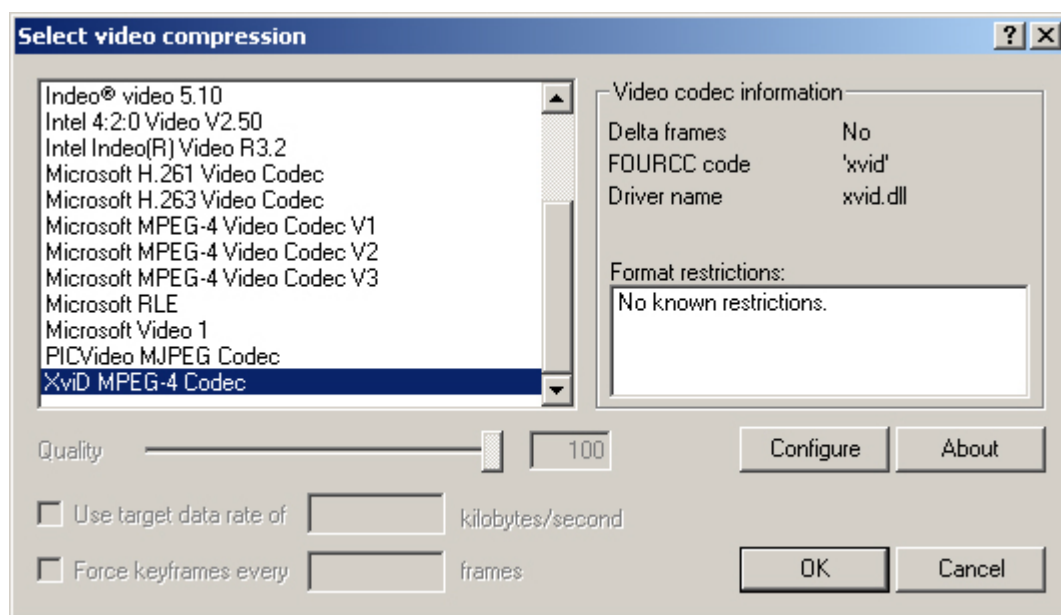
Some of these people wanted to continue the work, collected the latest sources they had and brought them together to continue with that work at <http://www.videocoding.de/>.

Since the start-over, XviD is developing very fast and is evolving into a very high quality codec.

You can download the latest binaries built from the CVS at <http://www.freewebz.com/xvid/> (Nic's page, URL may change, look at <http://forum.doom9.net/> and <http://www.xvid.org/> to find the actual download locations...)

This document explains the Video for Windows interface to XviD. It shows how to setup the codec to get best results from it. Since XviD is under heavy development, this document is updated very often, but the fundamental things should stay the same over the different versions.

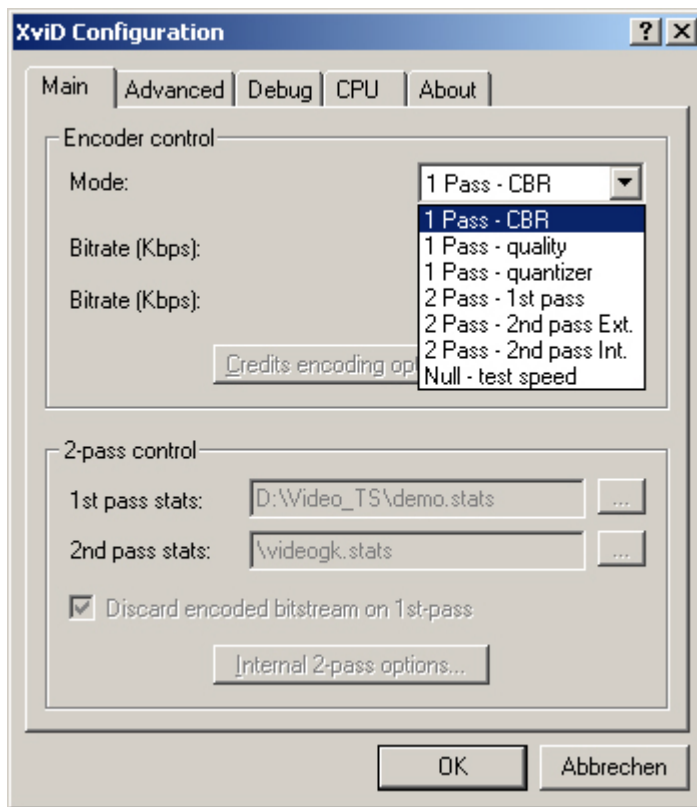
Using the codec



In your favorite encoding application (here: virtualdub, <http://www.virtualdub.org/>), go to the "codecs" menu. All usable codecs should be listed there. Now, after you have installed XviD successfully, it should show up as "XVID MPEG-4 CODEC" in this menu.

If it doesn't show up after you followed the steps describing the installation above, try to close your application and restart it. If it still doesn't show up, try rebooting your computer, now it should show up. If it still doesn't, start over with step 1 - installing the codec once more. Now choose the XVID codec and press "configure."

Setting up the XviD codec (VfW frontend)

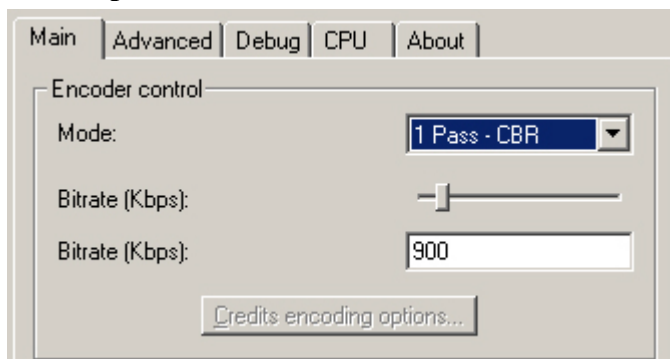


Now you will get presented the "Main" tab of the XviD VfW config interface.

Notice that dependant on the encoding mode you choose some controls will get enabled, disabled and/or renamed. This happens for two reasons:

firstly to save space, and secondly to make the interface easier to understand (earlier versions of the interface were cluttered and unorganised).

The "1 pass CBR" mode



This is the normal straight-forward mode you already know from most other codecs.

Choose a (average) bitrate that you want to target and encode at.

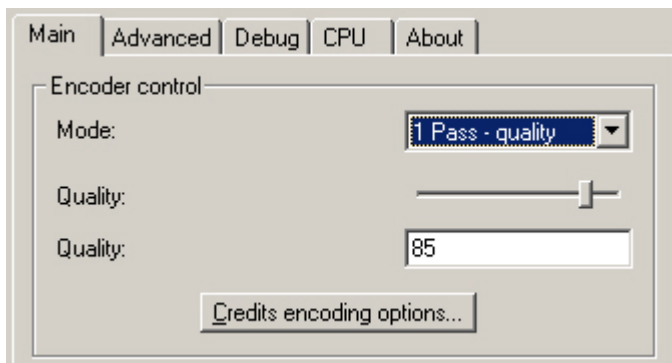
NOTE: at the moment, CBR mode does NOT work in combination with luma masking (discussed in later on).

So switch that off if you use 1 pass CBR!

Another side note, only in this place the value

of 1 kb is 1000 bytes (as requested by the MPEG4 specs), so keep that in mind when calculating your bitrate!

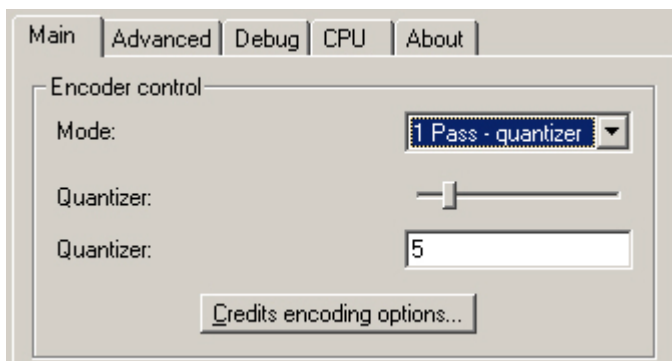
The "1 pass quality" mode



This mode is already known from DivX4. In this mode, all frames get the same amount of compression - regardless of the complexity of the frame.

This mode is useful if you want to reencode the material you're compressing this way.

The "1 pass quantizer" mode



This is another type of quality mode. If chosen, a fixed quantizer will get used, for example if you enter 3 here, all frames get compressed with quantizer (= level of detail removal) 3, resulting in a real constant quality. The difference to the 1 pass quality mode is, that here a fixed quantizer gets used, while in 1 pass quality the quantizer is modulated a little.

The "2 pass - 1st pass" mode

To produce the best quality encodes, you should use 2 pass mode. In this mode, the movie is processed twice. Choose this option for the first of the two passes.

You have to choose a location and name for a stats-file via the field called "2 Pass control", in which the statistical data gets collected.

During the first pass in 2pass mode the whole movie gets analysed with a fixed quantizer (2), to see how good the frames can get compressed. This information is used in second pass to scale down the quantizer/framesize to reach the desired filesize at a (usually high) constant quality level.

Don't change any of the settings between a first and a second pass which affect the compressability of the movie, e.g. don't add filters or change resize settings. Since the data gets collected with the first settings, the new settings do result in another compressability, but the frames get scaled down with the first pass data... this results in a bogus situation, where the quality achieved isn't as nearly as good as possible, rendering the 2 pass encoding useless. Be warned! :)

The "2 pass - 2nd pass Ext." mode

When you are done with the first pass and don't trust the capabilities of the builtin scaling routines, you can use Gordan Knot to edit the first pass stats file. If you do so, you don't need further explanations here as you are a little advanced and should know what you're doing!

Just use the 2nd-pass stats field in the 2-pass control and open your generated, already scaled GKnot stats file. No further curve treatment is done by XviD if you choose this encoding mode.

The "2 pass - 2nd pass Int." mode

This encoding mode activates the internal curve treatment/scaling algorithms of the XviD Vfw interface. With this option selected, all the settings you can apply get used through the second pass. We'll get to these options later in this document.

The credits encoding options

Credits

Starting credits

Process credits at start of movie:

Credits start at frame no.:

Credits end at frame no.:

Ending credits

Process credits at end of movie:

Credits begin at frame no.:

Credits end at frame no.:

Credits rate reduction

Encode credits at % rate:

Encode I-frames with quantizer:

Encode P-frames with quantizer:

Encode starting credits to size:

Encode ending credits to size:

OK

If you have start credits in a movie from which you think you don't need the best quality, enter the first and the last frame accordingly into the starting credits edit fields after checking the "process credits at start of a movie"-box.

The same applies to the end credits, if you think that your end credits need less bits - which you can spend better on the main movie and it's quality- then check the "process credits at end of movie"-checkbox and enter the start frame and the last frame which you want to be in lower quality.

The % rate value means % of the bitrate spent on the movie. So if the movie has average 800kbps, the credits would get 80 kbps when set to 10%.

If you choose to encode the credits with a fixed quantizer, make sure you set that on first pass as well! Else your desired size won't be achieved!

If you want a fixed size for your credits, the codec will try to do it's best to achieve that size - but if you request it to make 2 MB out of 10000 frames you can get into trouble as the max quantizer is 31.... you get the idea :) This value is in kb, with 1kb=1024 bytes.

The internal 2 pass options

Internal 2-pass options

Internal 2-pass tuning

Desired file size (KBytes):

Keyframe boost %:

Curve compression

Bitrate payback delay (frames):

High bitrate scenes %:

Low bitrate scenes %:

OK

If you click on the "Internal 2 pass options" button, this window shows up. Enter the desired size of the movie (without sound) in kbyte (1 kbyte=1024 bytes) here.

With *keyframe boost %* you can give more bits to the keyframes, so if a keyframe would get scaled to 5000bytes without this option, at 20% it would get scaled to 6000 bytes.

This results in a higher quality keyframe and gives you the possibility of scaling it better to the movie's bitrate curve instead of "blindly" restricting the intraframe quantizers.

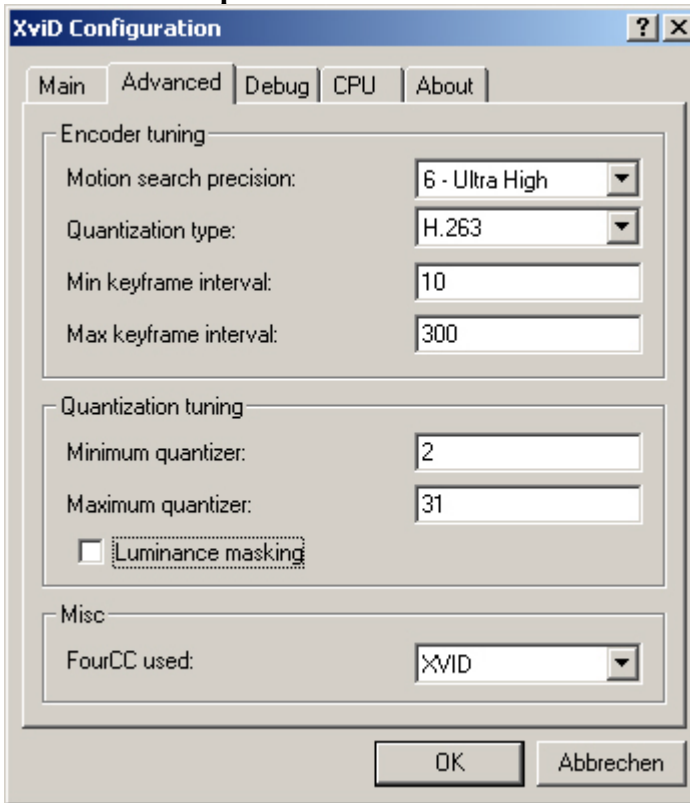
The *curve compression* is controlling the "smoothness" of the scaled bitrate curve. If you set curve compression low/high to 0, you'll disable it, having "perfect VBR" (you have to

set payback delay to 1 then as well).

Payback delay is the number of frames which should be used to compensate for under- or overuse of bits. Curve compression can be used to adopt the codec's behaviour to your needs: high motion scenes usually need more bits, and setting Curve compression high to 0 or lower will give those scenes more bits, while setting it higher as Curve compression low will make those scenes using less bits, taking care of the fact that you can't see details in high motion scenes as well as in low

motion.

The advanced options tab



Motion search precision

The *motion search precision* decides mostly over the resulting quality at a given bitrate. If set to low values, the codec will be faster at the costs of quality for the given bitrate. For usual encodings, like 2 CDs, "5-Very High" is enough. If you want the best possible quality, choose "6 - Ultra High" here. This will be ~10% slower, but give a little improvements.

To get a little more technical here: modes from 1 to 3 are using nearly the same settings internally for motion estimation. Starting with precision 4, the codec uses a half pixel interpolation to achieve more precise results. At precision 4, the whole macroblock gets a single motion vector which describes the movement between two images.

Starting with precision 5, XviD uses "inter4v motion vectors", which means that all 4 8x8 blocks from the 16x16 pixel macroblock get

their own motion vector. With motion search precision 6 the search is done more often and thus a little slower (~10%) but increases compressability a little more again.

Quantization Type

The *quantization type* should depend on your available bitrate. For high bitrates (2CDs...) MPEG quant type results in better quality, like a sharper image. If you use low bitrates, .h263 quant type is your choice.

If you're doing a 2pass encode, the settings should be the same for both passes. If you choose "Modulated" (only for 2nd pass! If used in first pass it'll result in MPEG quant type!), you have to check which quantizer will get used more often. If the average quantizer in 2nd pass is e.g. 3.7, you should use .h263 quantization type, if 2.5 it's better to use MPEG quant type on first pass.

(Well, this hasn't been tested much, it works either way, because the internal algorithms will compensate for the oversized/undersized frames. But the impact on quality is yet unknown).

Keyframe interval

The *maximum keyframe interval* should be something around 10-12times the FPS (frames per second) of your source. Since on playback you can only jump to keyframes (intraframes), it'll result in longer seeking times when using fast forward or backward because the deltaframes (interframes) can just be constructed by playing all the frames from the last keyframe up to the wanted frame.

Minimum keyframe interval will work the other way - to get rid of many consecutive keyframes in a row, you need this value - e.g. 10 means that keyframes can only be inserted after 10 deltaframes have been encoded.

Quantization tuning

With *quantization tuning* you can restrict the range of quantizers used. This might help increasing quality if you don't allow high quantizers - but low bitrate conditions get spread over a bigger time range then. This might result in breaking the size predictability, so use with care!

Luminance masking

Luminance masking is something like a "psychovisual model". It uses the fact that the human eye can't resolve many details on very bright and very dark regions and thus lowers the bitrate in those regions. Use with care as well, it seems it's doing more harm than good on high bitrate situations and if intraframe quantizer locking gets used (together with quantizer smoothing).

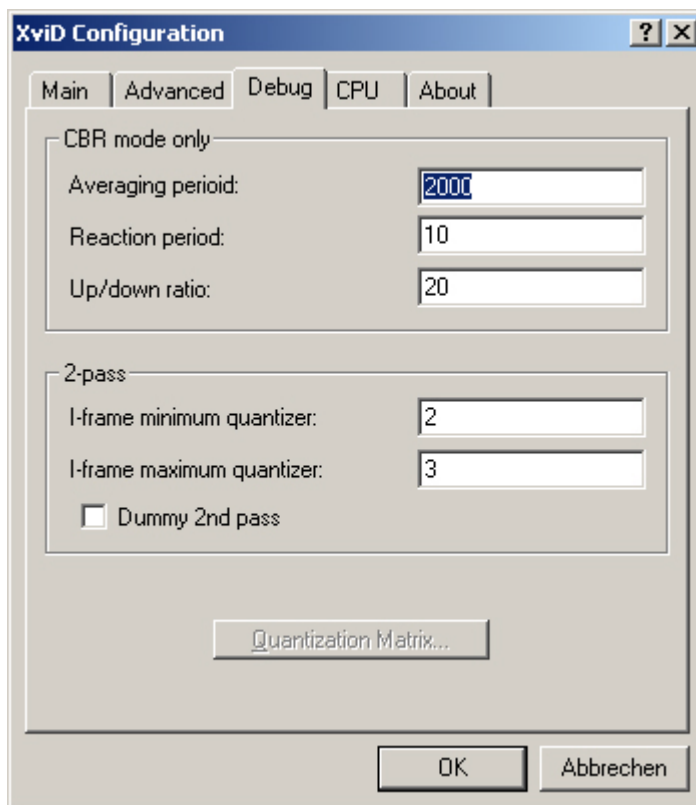
If you're going to do a 2pass encode, only use it on second pass and only if you don't use iframe quant lock! If you use it on the first pass as well, the bitrate reductions multiplies on scenes where the reduction kicks in, resulting in a very ugly overall image!

FourCC

The FourCC used is a temporary hack until we get our directshow filter to use post processing as well. If set to DIVX, the DivX4 playback directshow filter gets used, and you may experience degraded image quality since that DivX4 filter can't play the MPEG-quantization-type correctly. Most of the times the filter does too much filtering, rendering the image a little too smooth.

If you set FourCC to XVID, then the resulting avi will be played back with the XviD filters. Nic released a DS filter with brightness control and post processing, it's CPU intensive though. Install that and you can watch your XviD without other codecs installed :)

The debug options tab



1 pass CBR

As the box's title says, the first settings are used on *1pass CBR* mode only.

Averaging period describes the span of frames over which the codec could vary the bitrate so that the average bitrate is reached. The *reaction period* is the amount of frames in which the codec should stop the over-/underuse of bits.

The *up/down-ratio*.... well, still have no clue what it does precisely :)

2 pass options

Once more - as the title says, the *2pass* options only get used when you are in 2pass mode ;)

The *I-frame min and max quantizers* are the restriction for intraframes (keyframes). Those are image storing the whole data of an image. If these are high quality (=low quantizer), then the delta frames after those can look better, because the macroblocks which get shifted around are higher quality.

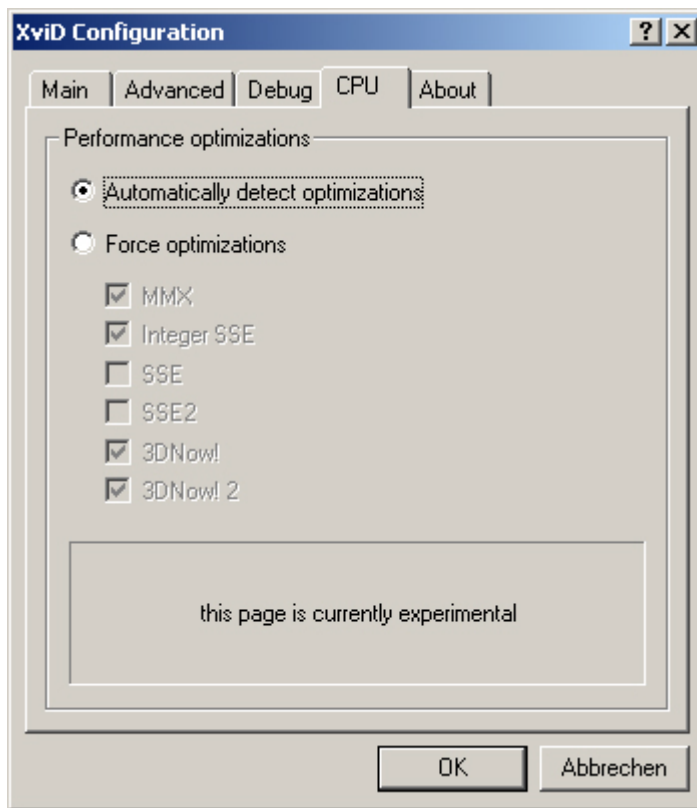
Use these settings with care. They can break file size predictability - and, if the settings are too strict, the codec tries to compensate for the overuse of bits, resulting in really bad quality deltaframes. Once more, this is experimental and not easy to decide, so use with care!

Quantization matrix

The *Quantization Matrix...* button is for user defined quantization matrices. The quantization matrix determines the divider used by quantization for each DCT coefficient. The DCT coefficients range from low-frequency values at the top left, to high-frequency at the bottom right. By entering lower numbers in the matrix, those coefficients will be quantized "less", and thus be closer to the original source. Higher numbers achieve the opposite - quantizing more, lowering quality, but lowering file size. So as you can see, the MPEG matrix is "biased" towards low-frequency coefficients.

The H.263 matrix has the same matrix value for all frequencies, low and high, which results in a more blurry image, because high and low frequency coefficients are quantized with the same weight.

The CPU options tab



Usually, you should leave this on auto detection. If you get weird crashes or strange results, you might first try to disable SSE2+SSE2, if still nothing changed disable 3DNow+3DNow2... and so on.

This is only for situations where the wrong CPU extensions get used and produce erroneous output.

The about tab



Well, nothing big to explain here... you find a link to the home of XviD - and some kind of build-information...

For example, when the XviD version you're using right now was compiled :)

In the near future there will be as well shown who built (or modified) the version you're using, so you can easily send in bug reports that are more useful for us.

The XviD Directshow Filter



This is the actual XviD Directshow Filter coded by Nic.

With the brightness control you can make too dark movies better watchable.

The post-processing settings are as following:

Horizontal deblock (Y) is deblocking on the horizontal axis in luminance space, vertical deblock on the other axis.

Horizontal deblock (C) is the same in chrominance space (U+V), vertical deblock accordingly.

Deringing(Y) is deringing in luminance space again, and deringing (C) in chrominance space.

Using the filter in the luminance (Y) space is more visible for the human eye and should be chosen first, especially on slower computers.

You can combine it with filtering in chrominance space (C), but that gain in quality isn't that much as for the (Y) filters.

Last but not least the deringing is there to reduce "mosquito" noise. Since this code isn't optimized (yet) it needs much cpu-power and can lead to choppy playback on slower machines.

—
EOD