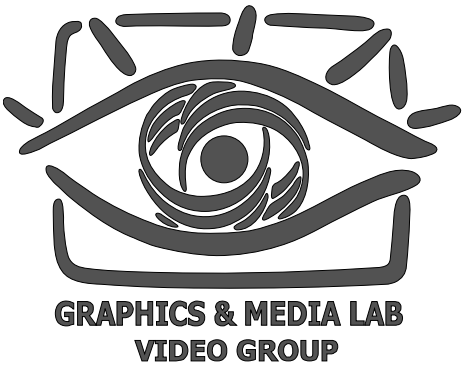


Lossless Video Codecs Comparison ‘2007

March 2007

CS MSU Graphics&Media Lab

Video Group

http://www.compression.ru/video/

videocodec-testing@graphics.cs.msu.ru

http://www.compression.ru/video/

videocodec-testing@graphics.cs.msu.ru

MSU Lossless Video Codecs Comparison ‘2007

Project head: *Dr. Dmitriy Vatolin*

Testing, analysis: *Iwan Seleznev*

Verification: *Dr. Maxim Smirnov*

Contents

1 List of Figures

2 Overview

2.1 Purpose

The main goal of this document and the performed comparison is getting answers on the following questions regarding lossless video codecs:

- What codec or codecs are best for video capture and video editing applications?
- What codec or codecs achieves the best compression ratio?
- What advantages multithreading gives to modern codecs supporting it?

Only **absolutely lossless** codecs were studied in this comparison. Only progressive test video sequences were used.

2.2 Comparison Rules

- Codecs were tested in three color spaces - RGB24, YUY2, YV12 - using three sets of generally the same sequences. All necessary transforms were done in VirtualDub.
- For each color space only absolutely lossless codecs were compared. The checks on absolutely lossless decoding were done beforehand using MSU Video Quality Measurement Tool (MSU VQMT)[1].
- Several encoding presets were used for different codecs according to the amount of available codec's options. Generally three presets were used to cover range of possible options: "Default", "Fast", "Maximum compression" ("Max" for short).
- Speed was measured for one run only. Potentially it might give noticeable errors in relative performance of very fast codecs for short video streams.
- Only encoding speed was estimated. For each codec decoding speed is at least not lower than its encoding speed.
- For x264 codec AVI files with the source video were modified so that instead of FOURCC code "DIB " they contained "YV12" (otherwise the codec didn't work), thus it didn't change the source AVI files size.
- For x264 codec resulting h264 files were converted to AVI files.

[1] http://www.compression.ru/video/quality_measure/video_measurement_tool_en.html

- Codec's compression ratio of a single sequence was calculated separately for each color space as the ratio between the sizes of the AVI files of a given sequence before and after compression.

- Codec's compression ratio for a given color space was calculated as an average of compression ratios of all sequences for this color space. So it does not depend on sequences' sizes generally, but does on their features and their handling a codec.

2.3 List of Tested Lossless Codecs

The tested codecs are listed in Table 1 below.

Table 1. List of tested lossless video codecs

Codec	Producer/ Author	Version	URL
Alpary	Alparysoft	v2.0 build 957.040607	www.alparysoft.com
ArithYuv	Julien Muchembled	v1.1.1	www.free-codecs.com/download/Arithyuv.htm
AVIzlib	Kenji Oshima	v2.2.3	www.geocities.co.jp/Playtown-Denei/2837/
CamStudio GZIP	RenderSoft	v1.0	www.rendersoftware.com
CorePNG	Jory Stone	v0.8.2	www.corecodec.org/projects/corepng
FastCodec	VideoSoft	v1.0b	videosoft.org
FFV1	M. Niedermayer	ffdshow 27.11.06 rev589	http://sourceforge.net/projects/ffdshow/
Huffyuv	Ben Rudiak-Gould, ffdshow team	v2.1.1 & ffdshow 27.11.06 rev589	http://neuron2.net/www.math.berkeley.edu/benrg/huffyuv.html http://sourceforge.net/projects/ffdshow/
Lagarith	Ben Greenwood	v1.3.12	http://lags.leetcode.net/codec.html
LOCO	M. Rezaei	v0.2	http://212.9.224.7/~video/CODECs/LOCO.lossless.CODEC.v0.2.exe
LZO	Abraham Macias Paredes	v0.4	http://usuarios.lycos.es/lzocodec/
MSU Lab	MSU Graphics & Media Lab	v0.6.0	www.compression.ru/video/ls-codec/index_en.html
PICVideo	Pegasus Imaging Corporation	v2.10.0.29	www.pegasusimaging.com/picvideolossless.htm
Snow	Michael Niedermayer	ffdshow 2006-08-02 rev127	http://sourceforge.net/projects/ffdshow/
x264	x264 team	rev604	www.videolan.org/developers/x264.html
YULS	YUVsoft	v1.0.3	www.yuvsoft.com/download/lossless-codec/index.html

2.4 Short Test Set Description

Table 2 shows brief information on the test set used in our comparison. For each sequence its RGB24, YV12, and YUY2 version was created or taken. On the whole, however, all videos were originally from YUV 4:2:0 or YUV 4:2:2 sources. All sequences are progressive. Section “Test Set” below gives more details concerning video streams selection and their specific characteristics.

Table 2. List of video sequences from the used test set

Sequence	Number of frames	Resolution
foreman	300	352x288
susi (di)	374	704x576
tennis (di)	373	704x576
bbc (di)	374	704x576
battle	1599	704x288
news	32	720x480
da	262	720x352
mi	261	640x272
bankomat (di)	376	704x352

3 Experiments

3.1 Measurements

Several points should be mentioned to give the proper understanding of measurement methods we used.

- To get sets of sequences in all color spaces, a color space tranforms were done in VirtualDub 1.6.17 from the original color space of a sequence to the other two.
- For codecs with VFW interface (actually those are all codecs except command-line x264 codec) were run via VirtualDub 1.6.17.
- To ensure that a given codec is lossless, the professional version of MSU VQMT[2] was used.
- For changing “DIB ” FOURCC code sequence to “YV12” in AVI files for x264 Hex editor tool from VirtualDub was used.
- For converting h264 files to AVIs avc2avi.exe tool was used.
- For automation of comparison process AutoIt v3 was used.

3.2 Environment

Tests were run on PC of the following configuration:

- Processor: Intel Centrino Duo T2400, 1.83 GHz
- Memory: 1 Gb
- Operating System: Windows XP Pro

For multithreading analysis additional tests were done using an IBM IntelliStation Zpro server of the following configuration:

- Processor: 2x Xeon Nocona 3.4 GHz
- Memory: 1 Gb
- Operating System: Windows XP Pro

[2] http://www.compression.ru/video/quality_measure/video_measurement_tool_en.html

3.3 Test Set

3.3.1 Principles of Test Set Selection

Nine sequences were used in the comparison, including some “standard” sequences and sequences with some special features, like noise, lots of flashes and scene changes.

The limitations are that all videos were originally from YUV 4:2:0 or YUV 4:2:2 sources. All sequences are progressive.

Here is a brief description of sequences.

3.3.2 Foreman

Sequence title	**foreman**
Number of frames	**300**
resolution	**352x288**
Source	**Uncompressed progressive (standard sequence)**

Frame 77

Frame 258

This is one of the most famous sequences. It represents a face with very rich mimic. Motion is not very intensive here, but on the other hand it is disordered, not straight-forward. Intricate type of motion creates problems for the motion compensation process. In addition, camera is shaking that makes the image unsteady. In the end of the sequence camera suddenly turns to the building site and there follows an almost motionless scene. So this sequence also shows codec’s behavior on a static scene after intensive motion.

3.3.3 Susi

Sequence title	**susi (di)**
Number of frames	**374**
resolution	**704x576**
Source	**MPEG-2 (40Mbit), Smart Deinterlace**

Frame 193

This sequence is characterized by relatively high noise and slow motion. The first part of video is almost static (the girl only blinks), then there is some quite rapid motion (she abruptly moves her head) and then the scene becomes almost static again. Noise is suppressed on every second frame due to the B-frames option in an original MPEG-2 codec.

3.3.4 Tennis

Sequence title	**tennis (di)**
Number of frames	**373**
resolution	**704x576**
Source	**Standard sequence, Smart Deinterlace**

Frame 97

This sequence shows some ping-pong game and its features are quite complicated motion, camera zoom, and static background. There are also several scene changes.

3.3.5 BBC

Sequence title	**bbc (di)**
Number of frames	**374**
resolution	**704x576**
Source	**Uncompressed (standard sequence), Smart Deinterlace**

http://www.compression.ru/video/

Frame 185

Frame 258

This sequence is characterized by pronounced rotary motion. It contains a rotating striped drum with different pictures and photos on it. This video features a strong vertical motion generally non-typical for a common video.

3.3.6 Battle

Sequence title	**battle**
Number of frames	**1599**
resolution	**704x288**
Source	**MPEG-2 (DVD), FlaskMPEG deinterlace**

Frame 839

This sequence is a fragment of the “Terminator-2” movie and represents its very beginning. This video may cause problems for encoders because of relatively week correlation between frames. That is because of three main reasons: constant brightness changes (explosions and laser flashes, see the picture above), generally very quick motion and frequent scene changes. There are also noticeable lossy compression artifacts in this video sequence due to intensive changes in a scene.

3.3.7 News

Sequence title	**news**
Number of frames	**32**
resolution	**720x480**
Source	**TV tuner**

Frame 13

This sequence is a fragment of business news and is characterized by very high level of a specific TV-tuner noise. There is almost no motion, except running headlines.

3.3.8 DA

Sequence title	**da**
Number of frames	**262**
resolution	**720x352**
Source	**MPEG-2 (DVD), Deinterlacer**

Frame 73

This sequence is a fragment of “Devil’s Advocate” movie. The motion is pretty simple, however there are two scene changes and sequence contains rich mimic of an actress. It also features relatively heavy film grain noise.

3.3.9 MI

Sequence title	**mi**
Number of frames	**261**
resolution	**640x272**
Source	**MPEG-2 (DVD), Deinterlacer**

Frame 15

This sequence is a fragment of “Mission Impossible” movie. Except for actors’ mimic the motion is very simple, and most of the scene is static. There are also several scene changes. This is a typical, very common movie content.

3.3.10 Bankomat

Sequence title	**bankomat (di)**
Number of frames	**376**
resolution	**704x352**
Source	**MPEG-2 (DVD), Smart Deinterlace**

Frame 237

This sequence is a fragment of “Terminator-2” movie. Motion complexity is moderate. There are no scene changes. This is also a typical movie content.

3.4 Codecs

3.4.1 Summary of Codecs and Presets

Not all codecs appear to be fully lossless in all color spaces. The following Table 3 illustrates it.

Table 3. Support of absolutely lossless compression in tested codecs

Codec	RGB24	YUY2	YV12
Alpary	✓	✓	✓
ArithYuv	–	✓	–
AVIzlib	✓	–	–
CamStudio GZIP	✓	–	–
CorePNG	✓	✓	✓
FastCodec	✓	✓	–
FFV1	✓	✓	✓
Huffyuv	✓	✓	✓
Lagarith	✓	✓	✓
LOCO	✓	✓	✓
LZO	–	–	✓
MSU Lab	✓	✓	✓
PICvideo	✓	–	–
Snow	–	–	✓
x264	–	–	✓
YULS	✓	✓	✓

In general, up to three presets were used for each codec, depending on the number of parameters relevant to compression and speed. Presets can be divided into three classes: “Default” (balanced), “Maximum compression” (“Max” for short) and “Fast” one. The following Table 4 shows all presets for all codecs, at that parameters directly influencing neither speed, nor compression ratio are omitted.

Table 4. Presets of codecs

Codec	Available relevant parameters	Presets		
		Default	Max compression	Fast
Alpary	lossless compression:	<no such preset>	On	On
	enable prediction:		On	On
	compression:		Max	Realtime
	SSE acceleration enable:		On	On
Arithyuv	No any relevant parameters.		<no such preset>	<no such preset>
AVIzlib	Compress Mode:	<no such preset>	Hi Compress	Hi Speed
	Video Format(24bit Only):		RGB24	RGB24

http://www.compression.ru/video/

Codec	Available relevant parameters	Presets		
		Default	Max compression	Fast
	Multi Tread:		Off	Off
	PNG Filter:		On	Off
CamStudio GZIP	Algorithm for Compression:	<no such preset>	GZIB	GZIB
	Level:		9	1
CorePNG	Compression Level:	1	9	1
	Enable Delta-Frames (unstable):	Off	On	Off
	Auto Delta-Frames:	Off	On	Off
	Keyframe Interval:	300	300	300
	Subtract:	On	On	Off
	Up:	On	On	Off
	Average:	On	On	Off
	Paeth:	On	On	Off
	None Filter:	On	On	Off
	All Filters:	On	On	Off
	No Filters:	Off	Off	No
FastCodec	RGB24 formats compression:	Absolutely lossless	<no such preset>	<no such preset>
FFV1	Coder type:	<no such preset>	AC	VLC
	Context:		Large	Small
	Keyframes:		300	300
HuffYUV	RGB24 (v2.2.1)	<no such preset>		<no such preset>
	RGB24 compression:		Gradient	
	YV12 & YUY2 (ffdshow rev589)			
	Predictor:		Median	
	Adaptive Huffman Tables:		On	
Lagarith	Enable Null Frames	On	<no such preset>	<no such preset>
	Use multithreading:	Off		
LOCO	RGB24		<no such preset>	<no such preset>
	RGB24 compression method:	RGB24		
	Max Loss Per Pixel:	0		
	YV12 & YUY2			
	RGB24 compression method:	YUY2		
	Max Loss Per Pixel:	0		
LZO	Compression Algorithm:	LZO	RTPPM	<no such preset>
	Key Frames:	64	300	
MSU	Quality:	Absolutely lossless	Absolutely lossless	Absolutely lossless
	Mode:	Max compression, slow decompression	Max compression, slow decompression	Maximize speed
	Advanced options:	FullSearch: Disabled		
PicVideo	Compress RGB24 using lossless pseudo YCrCb	<no such preset>	On	<no such preset>
SNOW	Quality:	100	100	100
	Maximum I frame interval:	300	300	300
	4 MV:	Off	On	Off
	Iterative Motion Estimation:	Off	On	Off

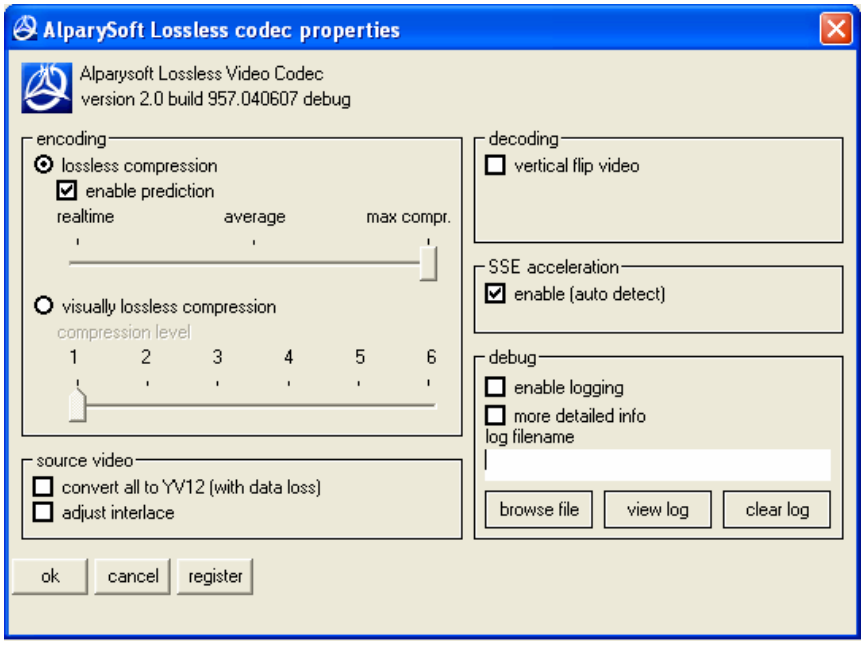
http://www.compression.ru/video/

Codec	Available relevant parameters	Presets		
		Default	Max compression	Fast
	ME compare function:	SSD, Chroma on	SSD, Chroma off	SAD, Chroma off
	Subpixel ME compare function:	SSD, Chroma on	SSD, Chroma off	SAD, Chroma off
	Macroblock compare function:	SSD, Chroma on	SSD, Chroma off	SAD, Chroma off
	Qpel refinement quality:	8	8	Off
x264		--qp 0 --keyint 300	--qp 0 --ref 8 --mixed-refs --keyint 300 --partitions all --me umh --subme 6	--qp 0 --keyint 300 --no-cabac --subme 1 --partitions none --no-chroma-me
YULS	Compression:	<no such preset>	Maximum Compression	Faster Compression
	Key frames:		300	300

In the following section brief description of codecs used in the comparison will be given and some specific problems and details will be mentioned.

3.4.2 Alpary v2.0

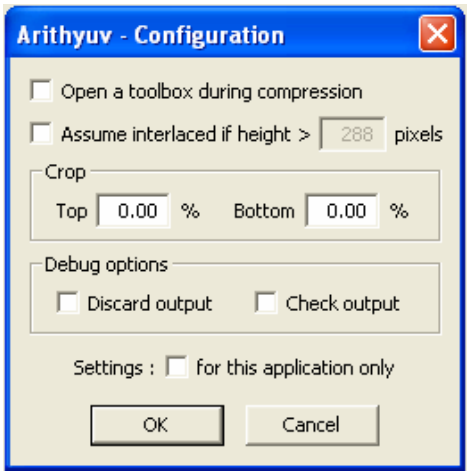
Supports lossless compression in RGB24, YUY2 and YV12 color spaces. Supports both VfW and DirectShow interfaces. Places a small company logo in the right bottom corner of a frame when unregistered, registration is free.



http://www.compression.ru/video/

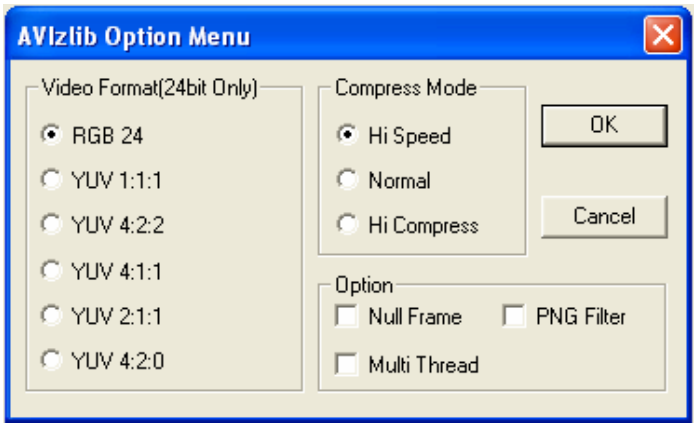
3.4.3 ArithYuv v1.1.1

Supports lossless compression in YUY2 only. No relevant compression parameters.



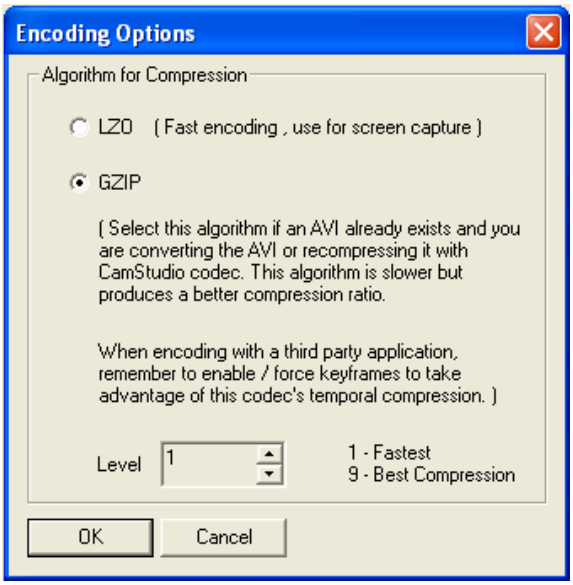
3.4.4 AVIzlib v2.2.3

Supports lossless compression in RGB24, is lossy YUV formats. For some reason isn't compatible with MSU VQMT. So to ensure absence of losses compressed files were decompressed and then measured. Supports multi-threading.



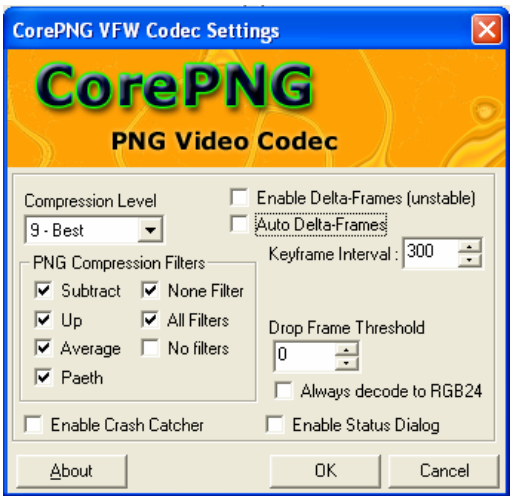
3.4.5 CamStudio GZIP v1.0

Supports lossless compression in RGB24.



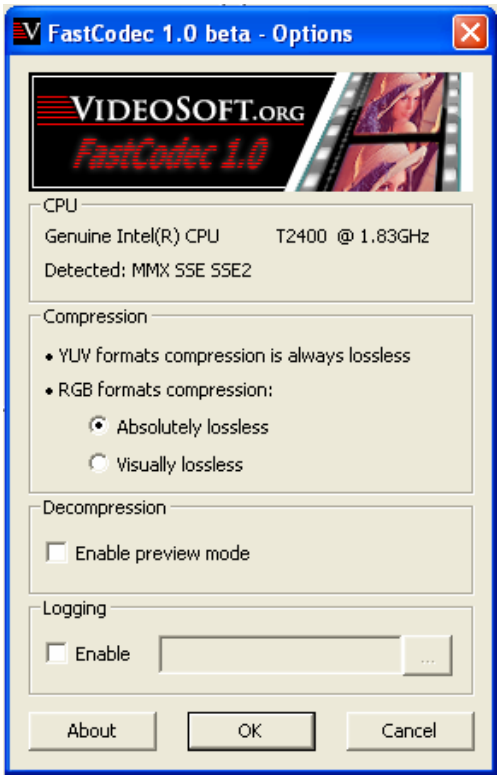
3.4.6 CorePNG v0.8.2

Supports lossless compression in all three tested color spaces.



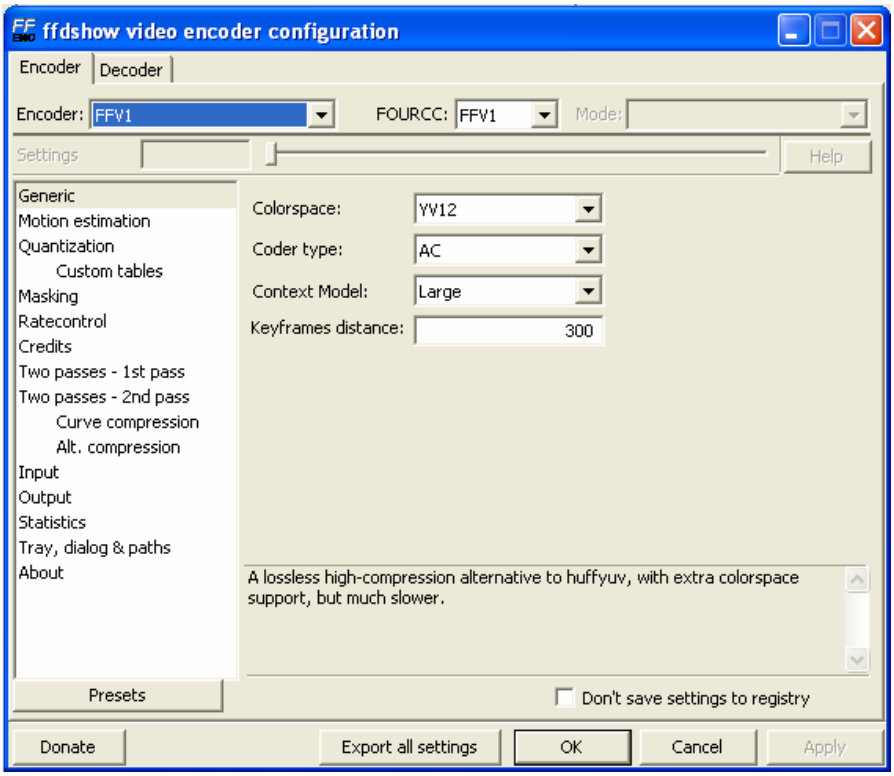
3.4.7 FastCodec v1.0b

Supports lossless compression in RGB24 and YUY2.


FastCodec 1.0 beta - Options
VIDEOSOFT.ORG
FastCodec 1.0
CPU
Genuine Intel(R) CPU T2400 @ 1.83GHz
Detected: MMX SSE SSE2
Compression
• YUV formats compression is always lossless
• RGB formats compression:
Absolutely lossless
Visually lossless
Decompression
Enable preview mode
Logging
Enable
About
OK
Cancel

3.4.8 FFV1 ffdshow rev589

Supports lossless compression in all three color spaces, however needs that color space mode was manually changed.

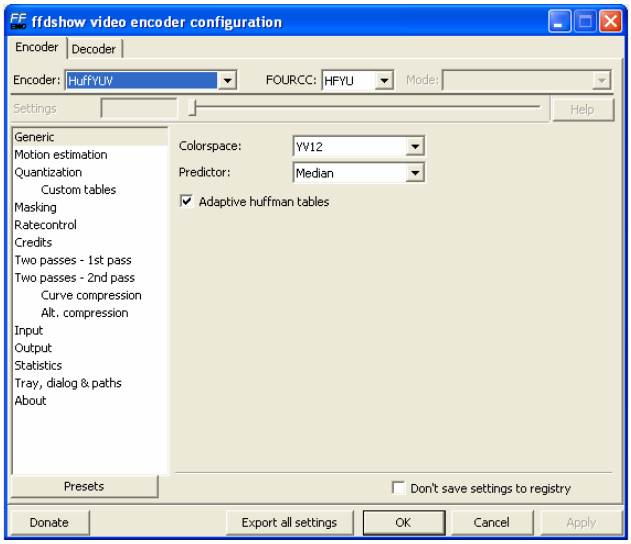


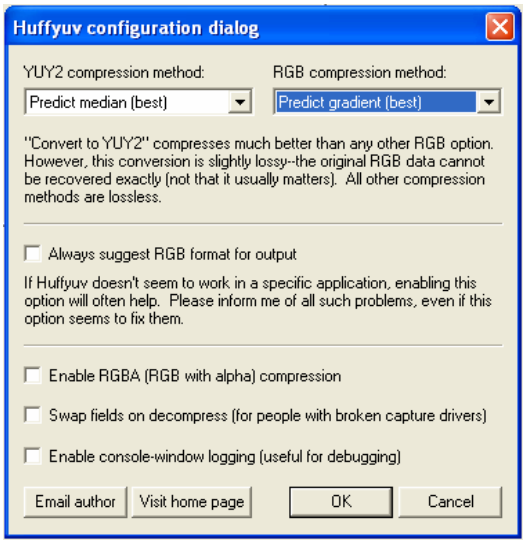
3.4.9 Huffyuv v2.2.1 & HuffYUV ffdshow rev589

The version from ffdshow supports lossless compression in YV12 and YUY2, and v2.2.1 supports it in RGB24 and YUY2. Two versions conflict with each other, so for proper use disabling of one of them may be needed. Huffyuv v2.2.1 was used both for RGB24 and YUY2 since it is a more common version, HuffYUV ffdshow was used for YV12 only.

It was found out that the maximum compression preset is usually also the fastest one.

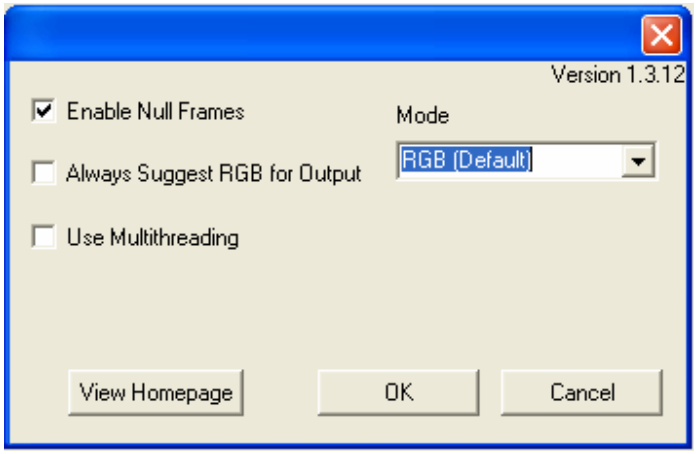
HuffYUV ffdshow implements a more advanced compression method with adaptive huffman tables than Huffyuv v2.2.1, so these are actually different codecs. However, for simplicity reasons the name ‘HuffYuv’ is used both for Huffyuv v2.2.1 and HuffYUV ffdshow in this report.


ffdshow video encoder configuration
Encoder
Decoder
Encoder: HuffYUV
FOURCC: HFYU
Mode:
Settings
Help
Generic
Motion estimation
Quantization
Custom tables
Masking
Ratecontrol
Credits
Two passes - 1st pass
Two passes - 2nd pass
Curve compression
Alt. compression
Input
Output
Statistics
Tray, dialog & paths
About
Colorspace: YV12
Predictor: Median
Adaptive huffman tables
Presets
Don't save settings to registry
Donate
Export all settings
OK
Cancel
Apply


Huffyuv configuration dialog
YUY2 compression method:
Predict median (best)
RGB compression method:
Predict gradient (best)
"Convert to YUY2" compresses much better than any other RGB option. However, this conversion is slightly lossy--the original RGB data cannot be recovered exactly (not that it usually matters). All other compression methods are lossless.
Always suggest RGB format for output
If Huffyuv doesn't seem to work in a specific application, enabling this option will often help. Please inform me of all such problems, even if this option seems to fix them.
Enable RGBA (RGB with alpha) compression
Swap fields on decompress (for people with broken capture drivers)
Enable console-window logging (useful for debugging)
Email author
Visit home page
OK
Cancel

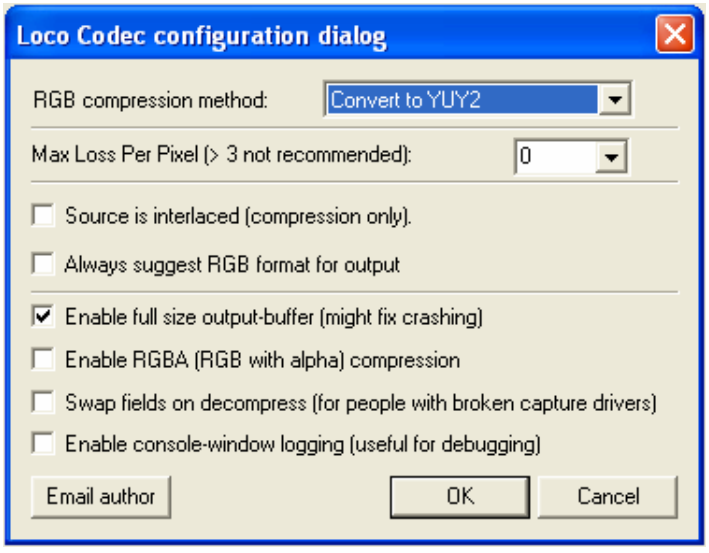
3.4.10 Lagarith v1.3.12

Supports lossless compression in all three color spaces, however needs color space mode be manually set in some cases: the codec processes YV12 video without conversion to RGB24 even in “RGB” default mode, but converts RGB to YV12 if YV12 mode is set. So, the mode controls conversion of “higher” resolution color spaces to “lower”. The codec supports multi-threading.



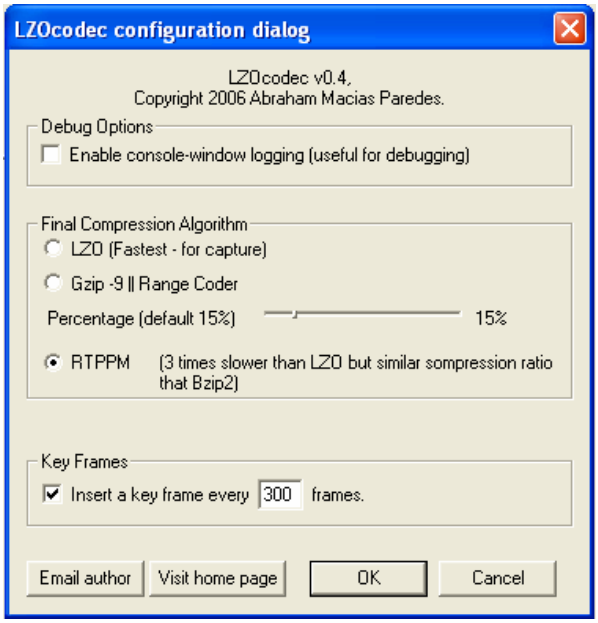
3.4.11 LOCO v0.2

Supports lossless compression in all three color spaces, for RGB24 processing “as RGB24” mode should be chosen.



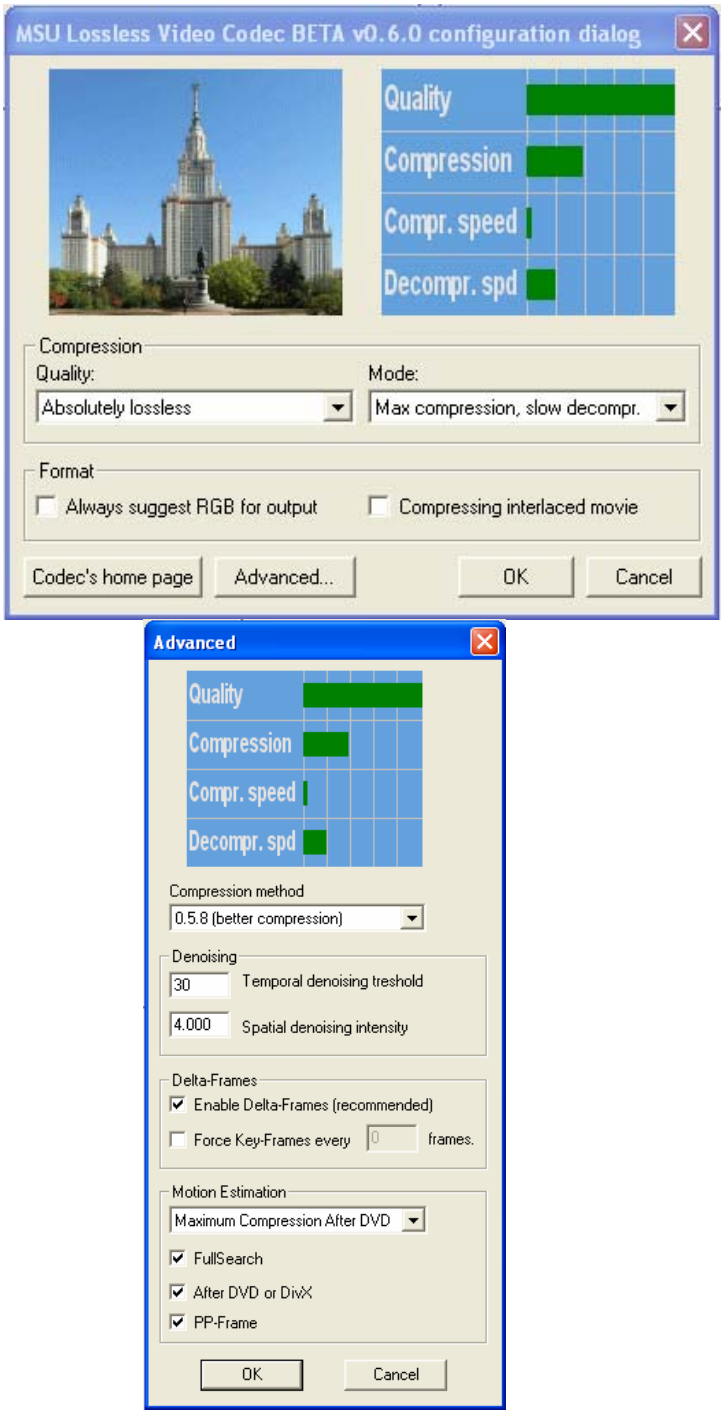
3.4.12 LZO v0.4

Supports lossless compression only in YV12, because converts all color spaces to this color space while decoding.



3.4.13 MSU Lab v0.6.0

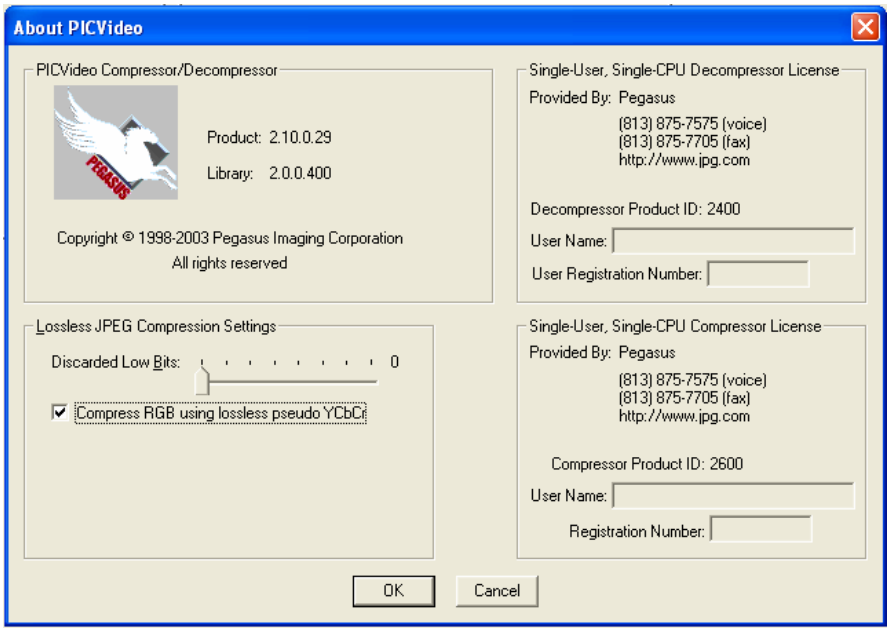
Supports lossless compression in all three color spaces.



http://www.compression.ru/video/

3.4.14 PICVideo v2.10.0.29

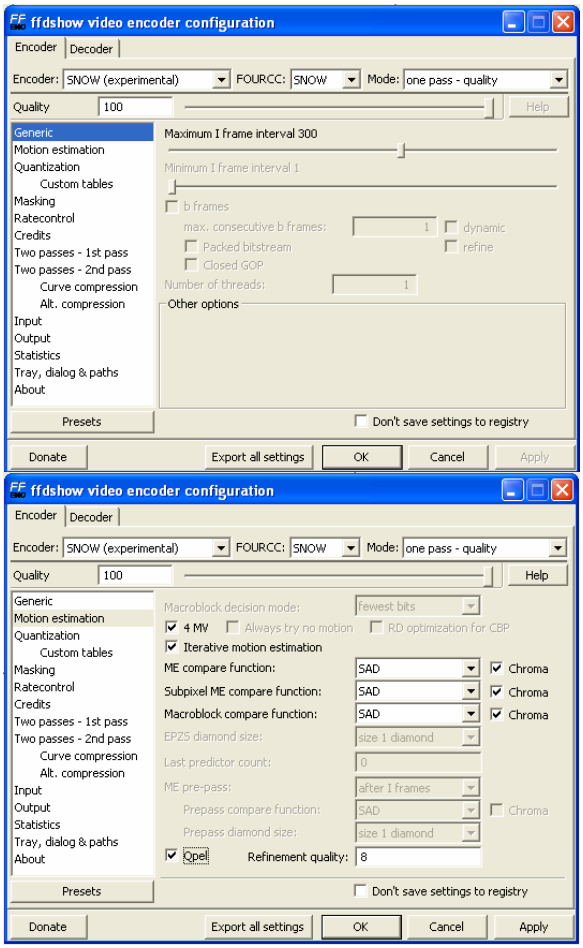
Supports lossless compression only in RGB24. Needs registration, otherwise inserts logo.



3.4.15 Snow ffdshow rev 127

Supports lossless compression only in YV12. For some reason was excluded from latest versions of ffdshow.

http://www.compression.ru/video/

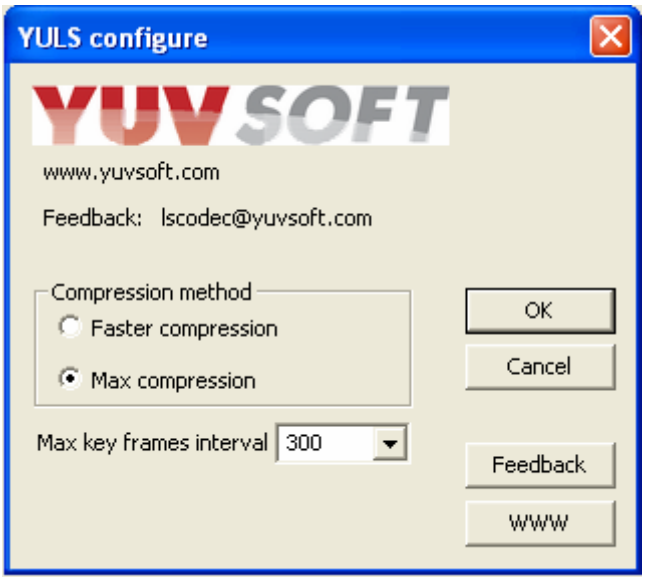


3.4.16 x264 rev604

Supports lossless compression in YV12. The codec has a command-line interface. For some reason it doesn't encode standard AVI files after VirtualDub color space transformation with "DIB " FOURCC sequence inside. So this char sequence should be manually replaced by "YV12" in any hex editor tool, for example the one in VD. It is obvious that it doesn't change original sequence size. Since all other codecs' compression ratios were measured as a ratio between sizes of two AVI files, in case of x264 resulting h264 file was converted to an AVI file using avc2avi.exe. This almost didn't change compressed file size, and made it possible to perform the lossless check using MSU VQM. Supports multi-threading.

3.4.17 YULS v1.0.3

Supports lossless compression in all three color spaces.



http://www.compression.ru/video/

4 Results and Analysis

This main section of the report is organized in the following way:

- firstly summary results for each color space are given and analyzed to choose codecs and presets to be compared in “Video capture and video editing” and “Maximum compression” competitions;
- more detailed results and study are presented for “Video capture and video editing” and “Maximum compression” areas respectively;
- additional data and conclusions on effectiveness of multithreading support goes next;
- finally global conclusions are made.

Those interested in the exact experimental data may refer to Appendix.

The report contains redundant and repeating data. Those interested in “Video capture and video editing” applications or, on the contrary, in “Maximum compression” might want to proceed to corresponding subsections at once after reading subsequent “Analysis Charts” subsection.

4.1 Analysis Charts

This technical subsection explains how charts presented in this report were made and gives guidelines on how one should interpret them.

- For speed comparison a 4CIF (or SIF, 704x576 pixels) frames per second (fps) unit of measurement was used. It is assumed that fps linearly depends on the amount of pixels in a frame. So 4CIF fps was computed in the following way:

***4CIF_fps* = *sequence_fps* * [(*sequence_height* * *sequence_width*) / (*704* * *576*)].**

- The average compression ratio and average 4CIF fps are calculated as the average of compression ratios and 4CIF fps respectively for each sequence in the test set.

- In trade-off charts 4CIF fps axes are directed to the left, so the more fps a codec provides the more to the left its marker is placed on a chart of this type.

- Compression ratio’s axes are directed upward.

- In speed / compression ratio trade-off charts the more to the left and to the up a codec is placed the better it is.

- In **ranking charts** each codec is ranked according to its result. For each file results of codecs are ordered and ranks are computed. A rank for the entire test set is computed as the average of per-file ranks. The better is the result (greater compression ratio, more fps) the lesser is its

rank. Rank axes are directed downward, so the better a codec is the more to the up it is placed.

- Y-axis values on the diagrams where one codec is compared to another (HuffYuv) are calculated as a ratio between the compression ratios of both codecs being compared.

4.2 Summary Results

Summary results are structured by tested color spaces: RGB24, YUY2, YV12. The internal structure of each paragraph related to RGB24, YUY2, YV12 is almost the same.

As it was mentioned, some codecs do not support absolutely lossless compression in some color spaces and do not some color spaces at all, so sets of codecs are different for each color space.

4.2.1 RGB24

The following graph on Figure 1 illustrates the trade-off between average compression ratio and average 4CIF fps for all codecs proved to be lossless in RGB24 and all their presets. The data used for creating this chart is given in Table 5.

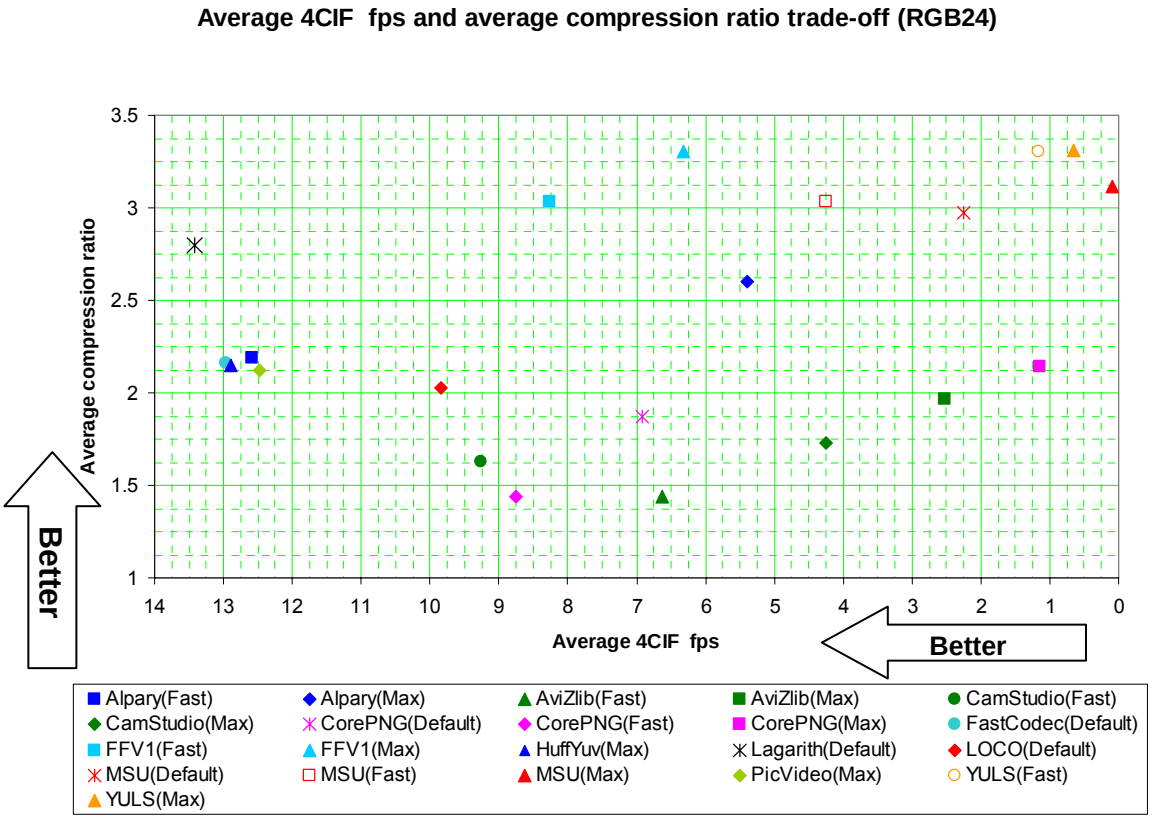


Figure 1. Average 4CIF fps and average compression ratio trade-off for RGB24 color space

Table 5. Average compression ratio and 4CIF fps data for RGB24

Codec	Average compression ratio	Average 4CIF fps
Alpary(Fast)	2,19	12,58
Alpary(Max)	2,60	5,39
AviZlib(Fast)	1,44	6,63
AviZlib(Max)	1,97	2,53

http://www.compression.ru/video/

Codec	Average compression ratio	Average 4CIF fps
CamStudio(Fast)	1,63	9,26
CamStudio(Max)	1,73	4,24
CorePNG(Default)	1,87	6,92
CorePNG(Fast)	1,44	8,75
CorePNG(Max)	2,14	1,14
FastCodec(Default)	2,16	12,97
FFV1(Fast)	3,03	8,25
FFV1(Max)	3,30	6,31
HuffYuv(Max)	2,15	12,89
Lagarith(Default)	2,79	13,43
LOCO(Default)	2,03	9,84
MSU(Default)	2,97	2,26
MSU(Fast)	3,03	4,26
MSU(Max)	3,11	0,08
PicVideo(Max)	2,12	12,48
YULS(Fast)	3,30	1,15
YULS(Max)	3,31	0,65

The next chart can be obtained from the previous one by uniting dots of the same codec into a single line.

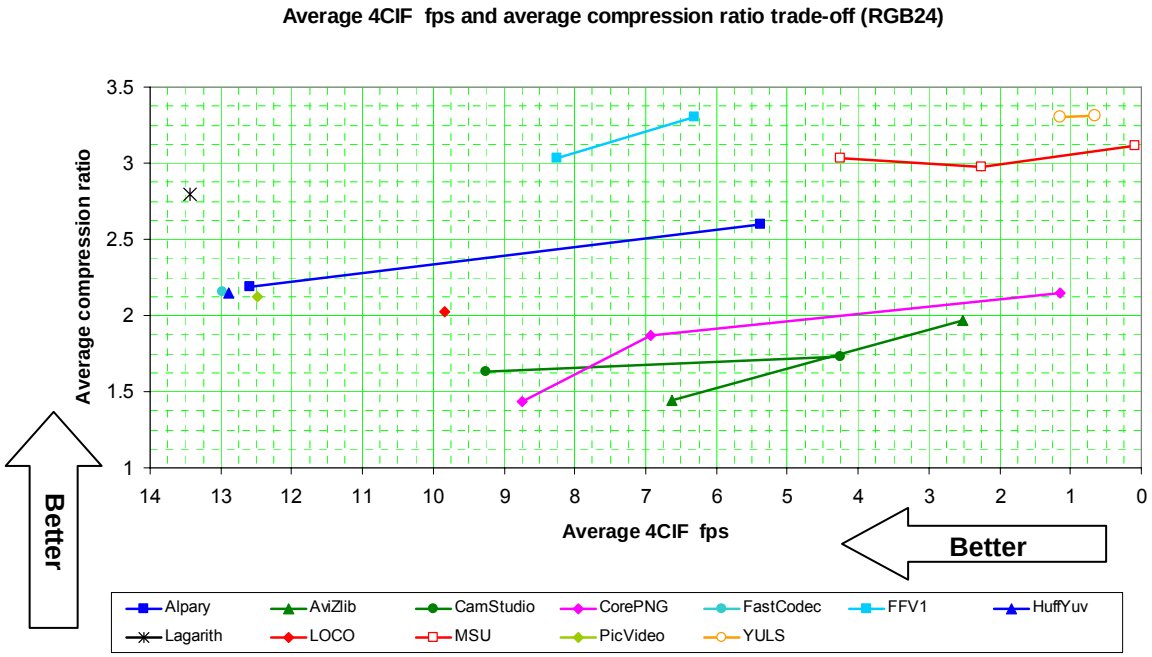


Figure 2. Speed-compression ratio curves for RGB24 color space

As it can be seen, maximum compression is provided by YULS, FFV1, MSU video codecs. At that YULS and MSU are generally the slowest codecs, MSU being the slowest for one of its preset. The fastest codecs are Lagarith, FastCodec and HuffYuv, there Lagarith is fastest for the used test set. The difference between FastCodec and HuffYuv is probably within an error of measurements and thus it should not be trusted requiring special check. Lagarith clearly beats relatively fast codecs both by compression ratio and fps.

The following six diagrams give more clear understanding of relative speed and compression ratio.

Figure 3 shows average compression ratio for codecs and presets ordered by decreasing of compression ratio.

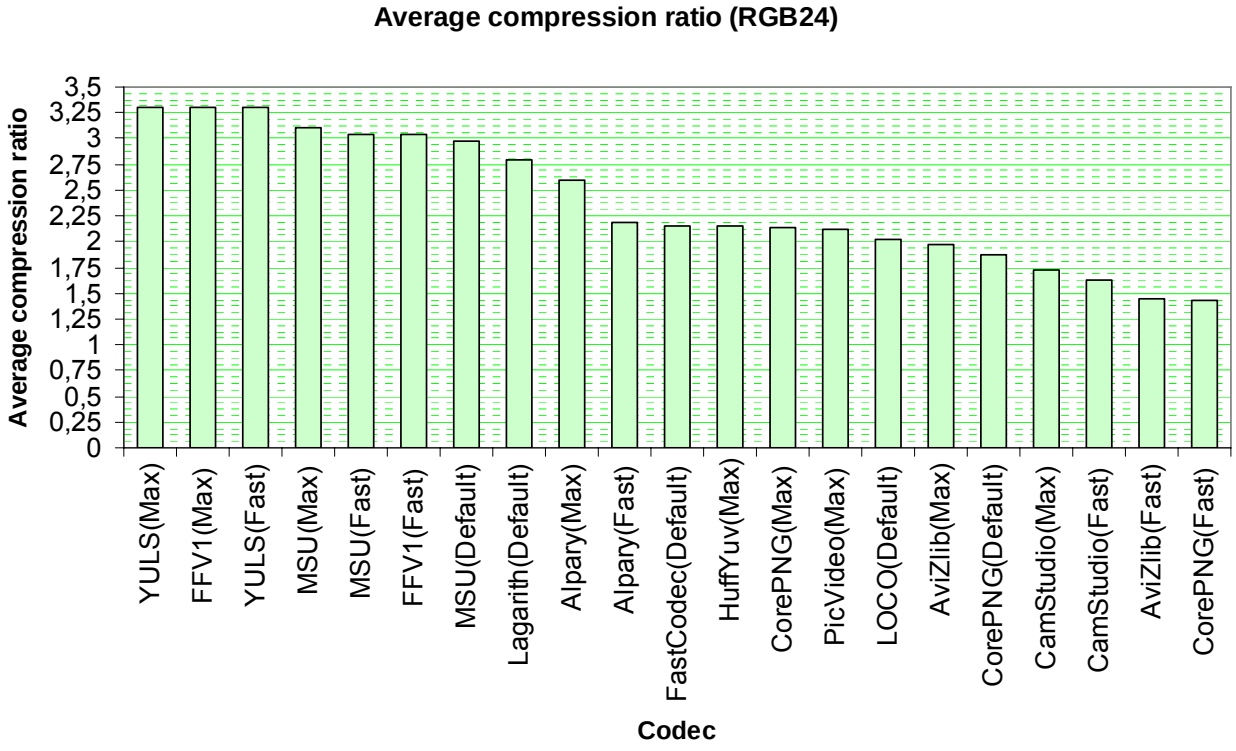


Figure 3. Average compression ratio for RGB24 for codecs and presets

The following diagram was obtained from previous by taking into account only best (in terms of compression) presets of each codec.

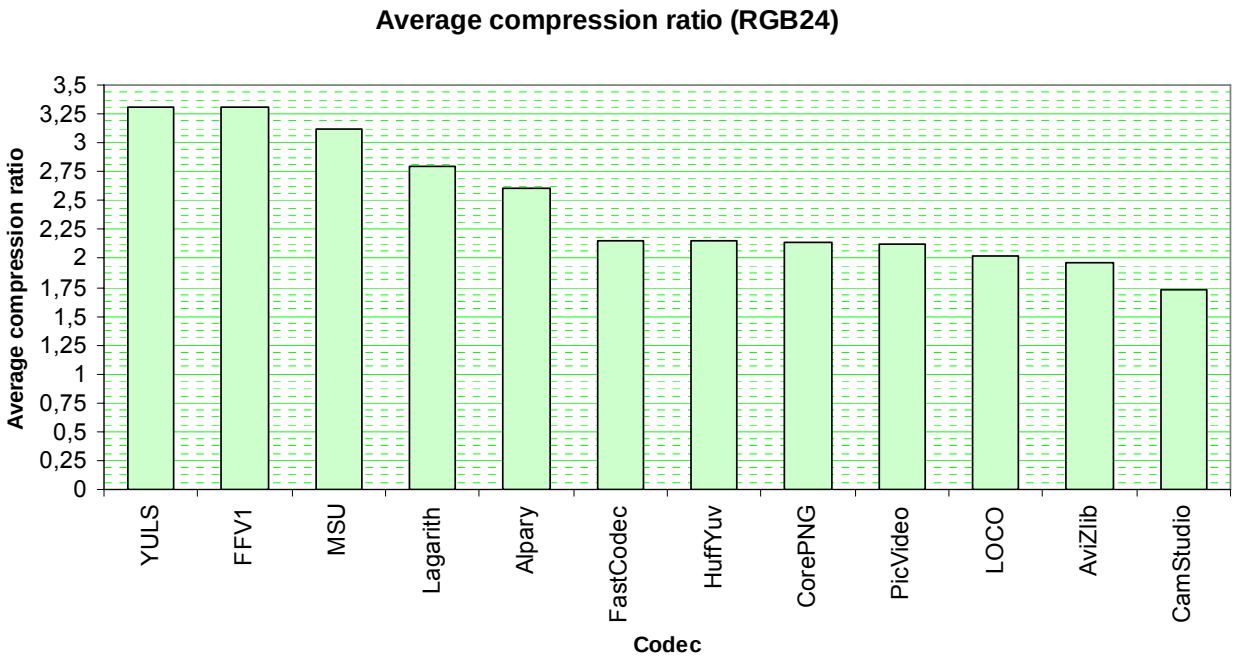


Figure 4. Average compression ratio for RGB24 for codecs only (best results)

An alternative to global ranking by average compression ratio is averaging of ranks obtained for each sequence (see also “Analysis Charts”). The next Figure 5 presents such data. On this graph the smaller a given value is the better a codec is, for this reason codecs are sorted in increasing order of the used metric.

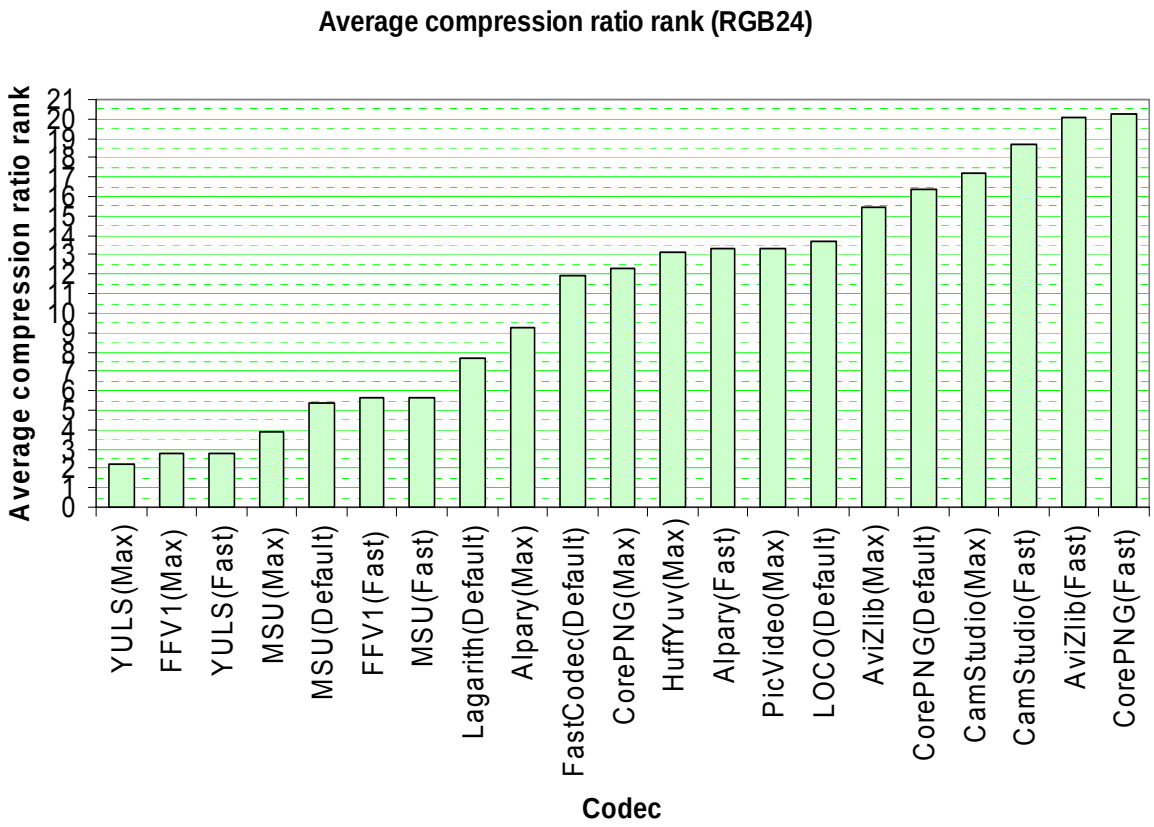


Figure 5. Average compression rank for RGB24 for codecs and presets

Figure 3, Figure 4, and Figure 5 highlight that the difference in compression ratio between the best compression codecs YULS and FFV1 is quite marginal for the chosen test set, YULS being best.

The following Figure 6, Figure 7, and Figure 8 give the similar data for fps. Figure 7 presents fps rate only for the fastest preset of each codec. Figure 8 shows relative position of codecs by encoding speed criterion than average ranks are used instead of average fps.

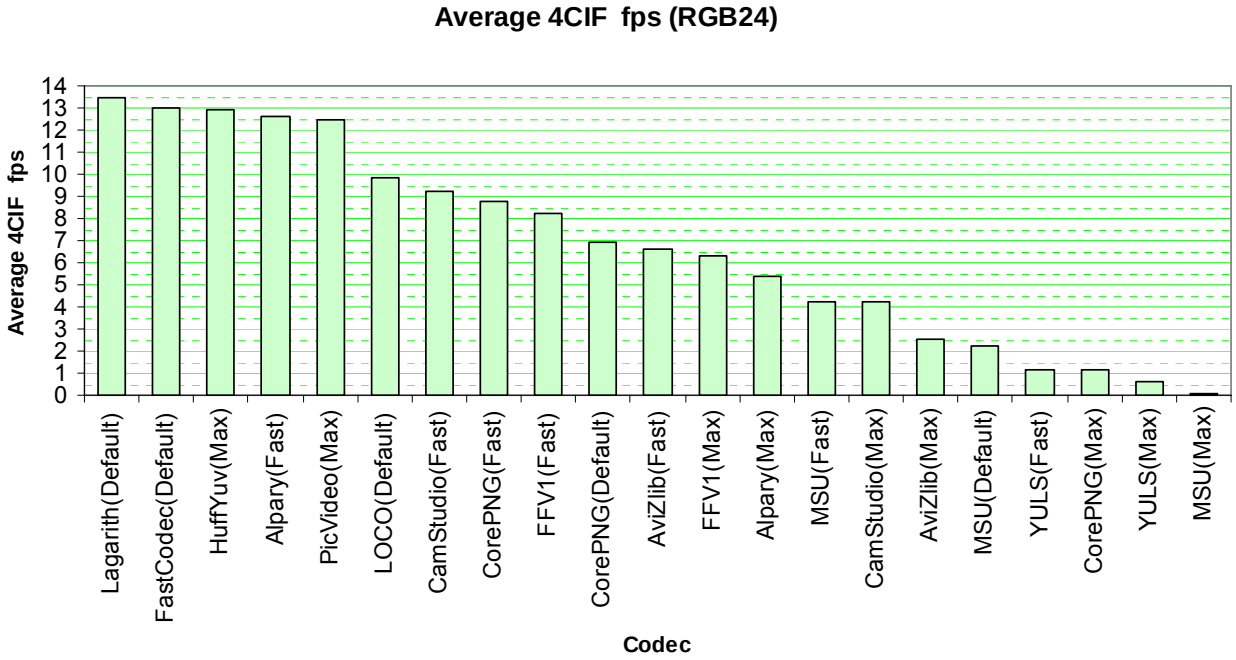


Figure 6. Average 4CIF fps for RGB24 for codecs and presets

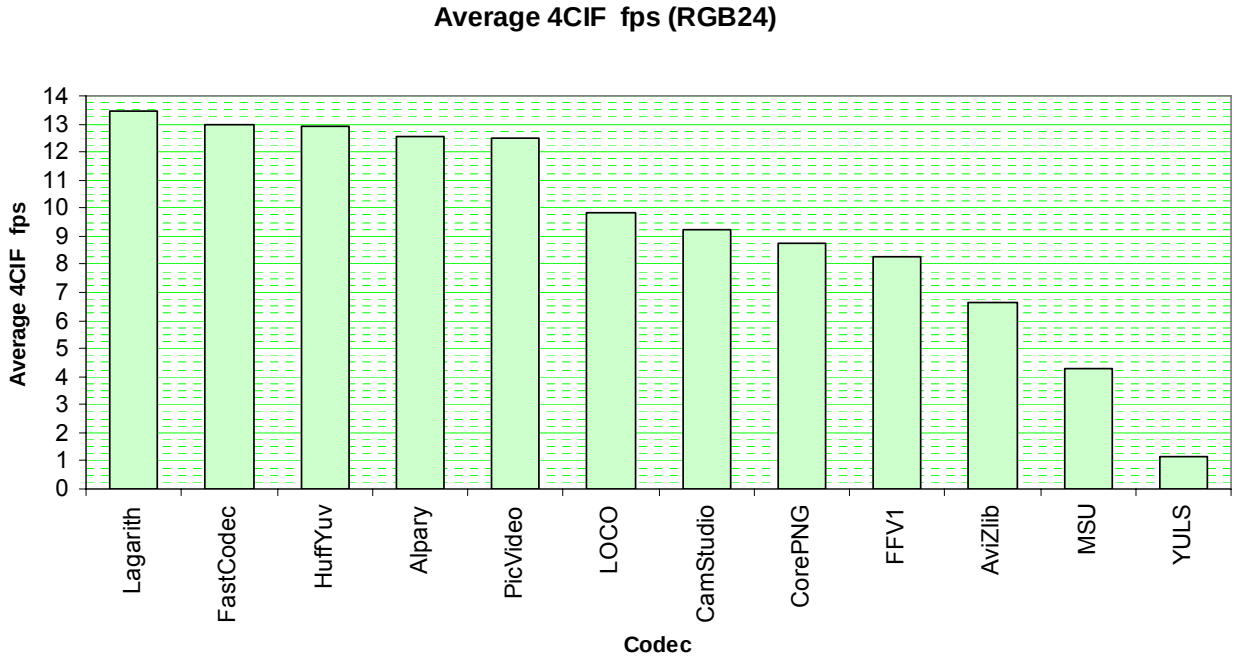


Figure 7. Average 4CIF fps for RGB24 for codecs only (best results)

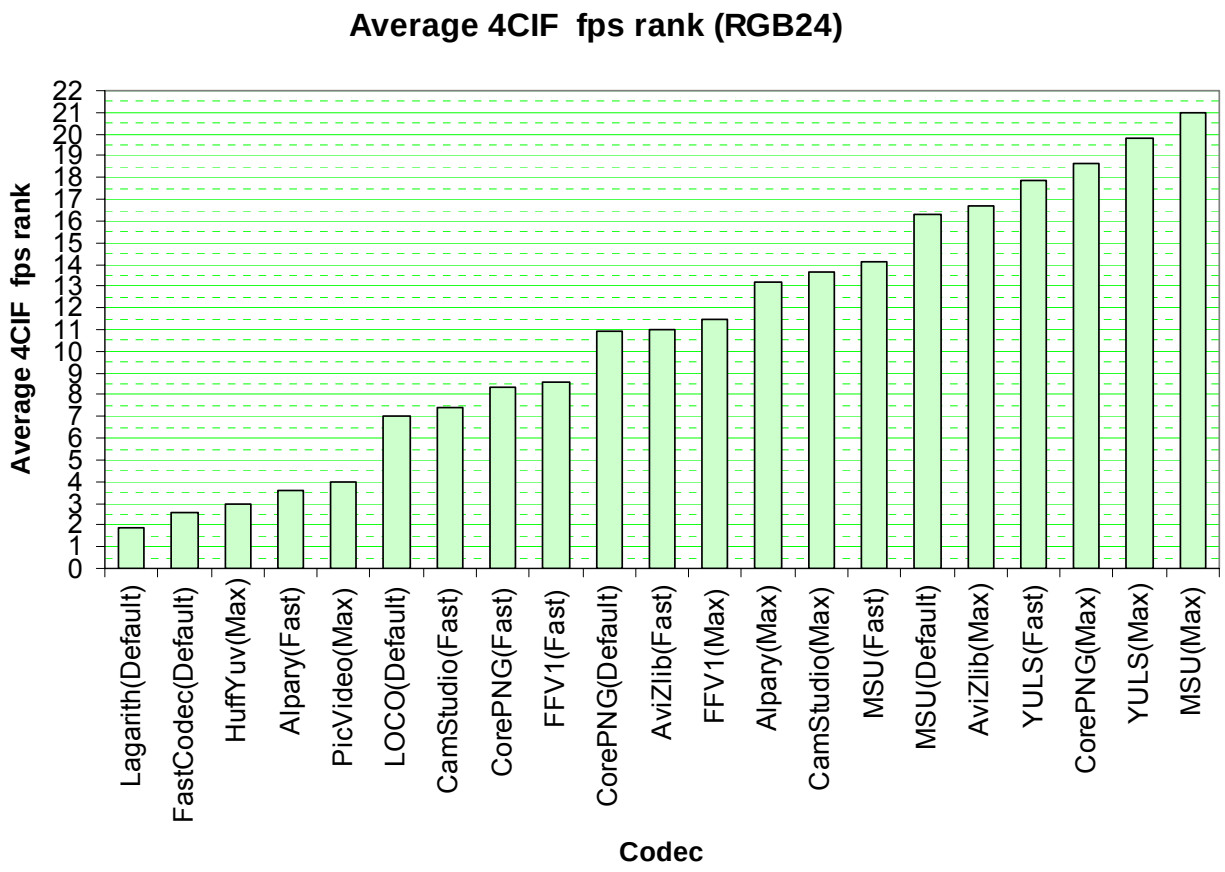


Figure 8. Average 4CIF fps rank for RGB24 for codecs and presets

As it was mentioned above, Lagarith shows the best encoding speed followed by FastCodec, HuffYuv, Alpary. The slowest are YULS, MSU (with its “Max” preset totally unpractical), “Max” presets of dictionary-based AviZlib and CorePNG codecs.

The following charts give insight in per-file data for codecs and their presets.

Figure 9 presents compression ratios for each sequence in the test set.

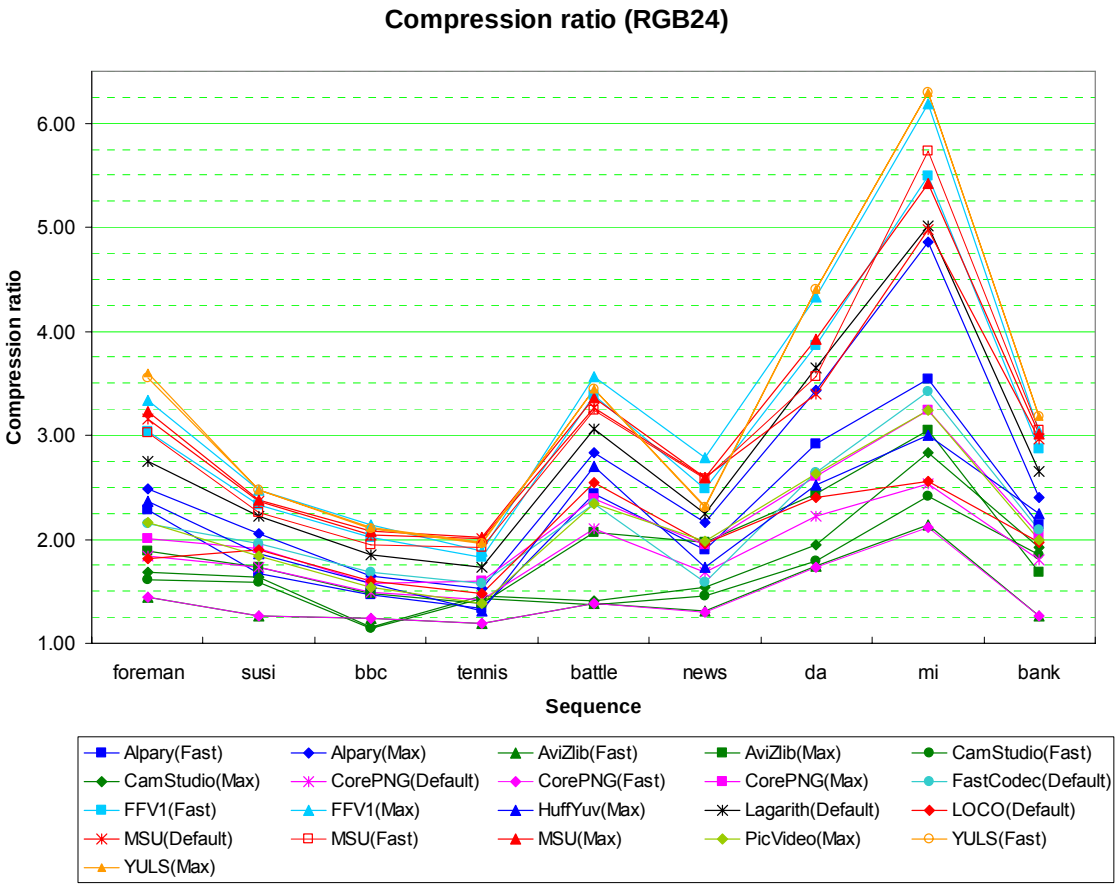


Figure 9. Per-file compression ratio for RGB24

From Figure 9 it is clear that YULS has a fault on “News” and FFV1 beats its competitors on “Battle” featuring video with comparatively week interframe correlation (quite untypical for common situations, it should be noted here). YULS surpass its rivals on “Foreman” classic sequence. CorePNG “Fast” and AVIZlib “Fast” which are based on the same compression algorithm, give almost exact results. These codecs and CamStudio consistently give the lowest compression ratio in general.

The following diagram shows only rating of compression ratio for each sequence. It helps to find out any unusual behavior and distinguish between codecs and presets of similar compression ratio.

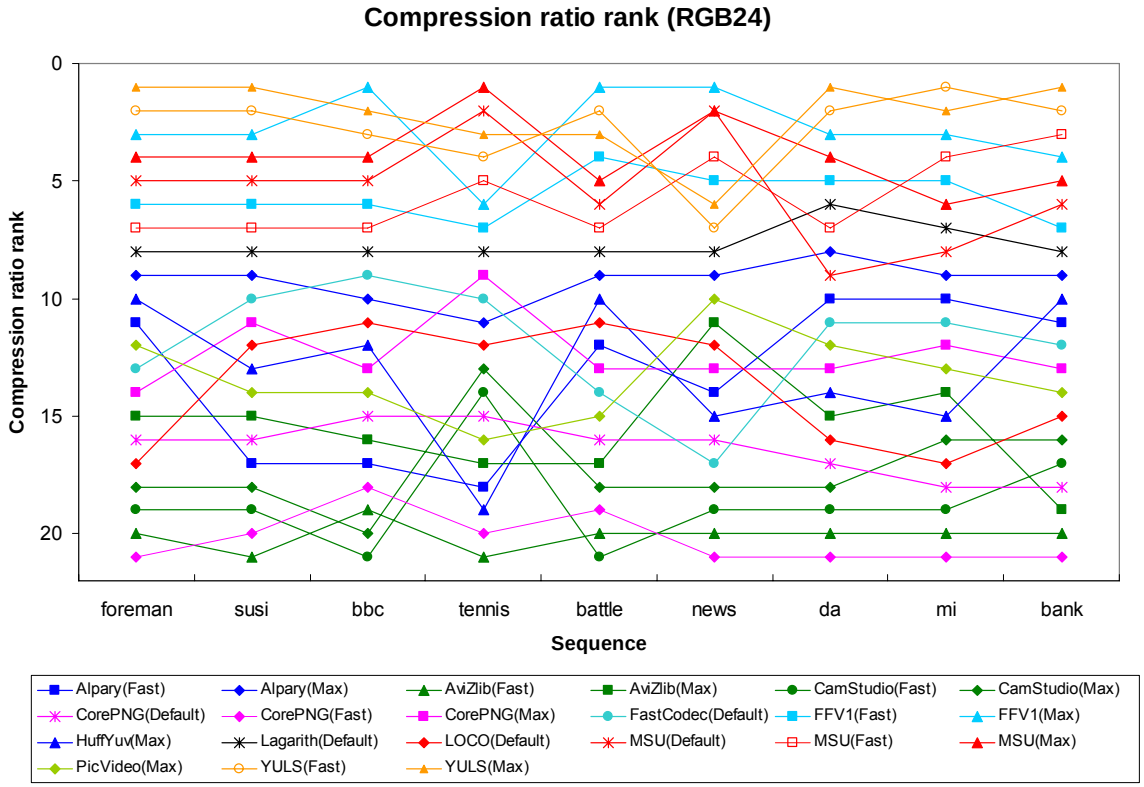


Figure 10. Per-file compression ratio ranking for RGB24

Analyzing Figure 10 it can be concluded that “Tennis”, “Battle” and “News” sequence are not so trivial to compress thus comparative results for these videos are somewhat different from the rest of the test set. The most predictable behavior can be seen for Lagarith, Alpary “Max” (always gives compression ratio above medium) and AviZlib “Fast”, CorePng “Fast” (almost always the worst compression). YULS, FFV1 always provide better results than Lagarith for the entire test set, and, as it can be seen from Figure 9, this lead is generally very noticeable. MSU falls behind Lagarith on movie sequences “Da” and “Mi” for some presets. “Tennis” is the only sequence where MSU is better than both YULS and FFV1. This sequence presents some problem to FFV1.

The next diagram shows relative compression ratio to distinguish codecs with relatively good and relatively bad compression abilities. As a reference competitor HuffYuv (Max) was chosen as a natural candidate.

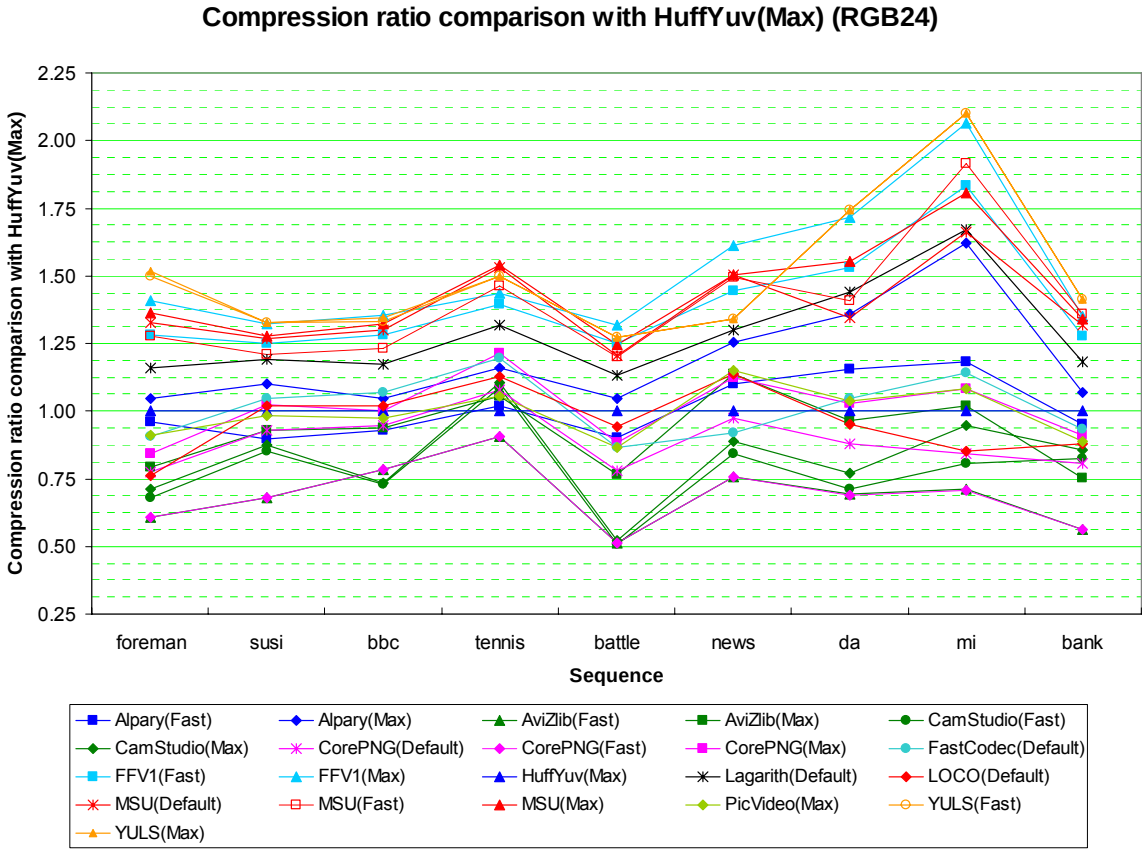


Figure 11. Relative per-file compession ratio for RGB24 in comparison to HuffYuv

The last two charts on Figure 12, Figure 13 show encoding speed data for every file of the test set. Despite normalization to 4CIF fps, generally there is a bigger deviance in codecs speed than it can be imagined.

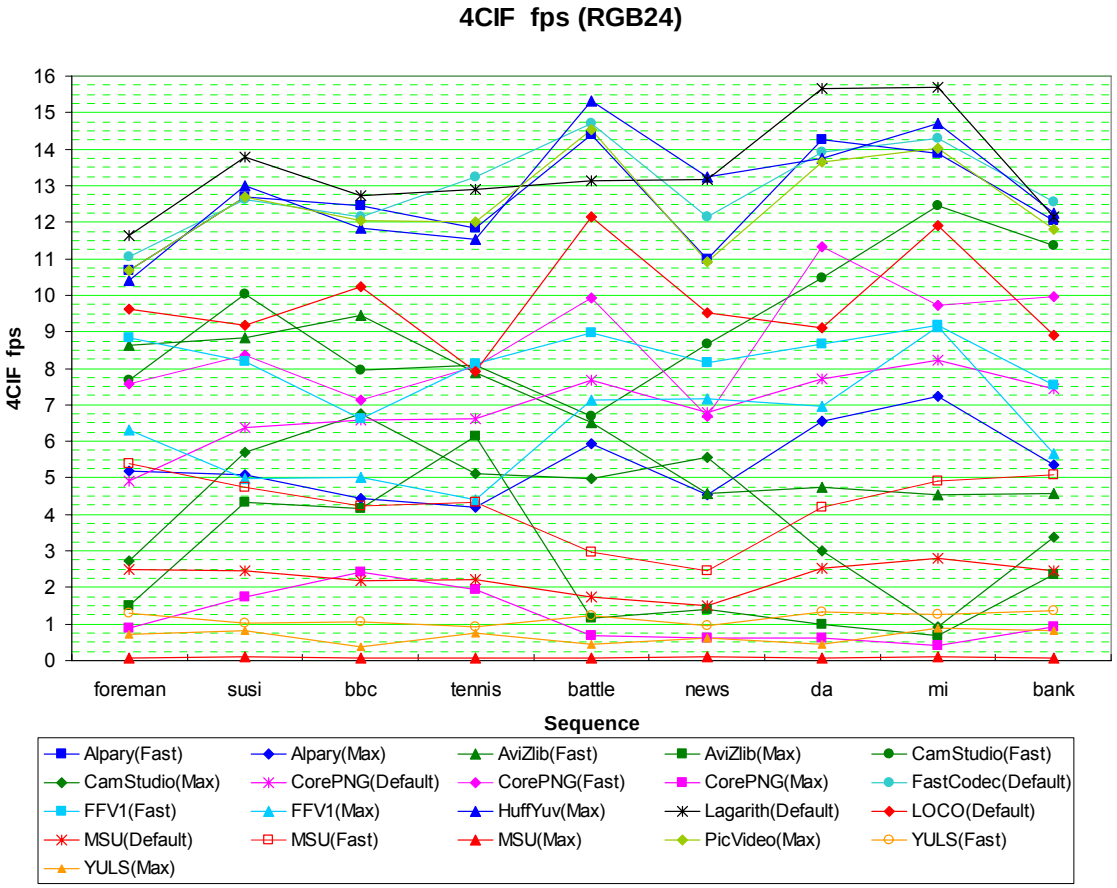


Figure 12. Per-file 4CIF fps for RGB24

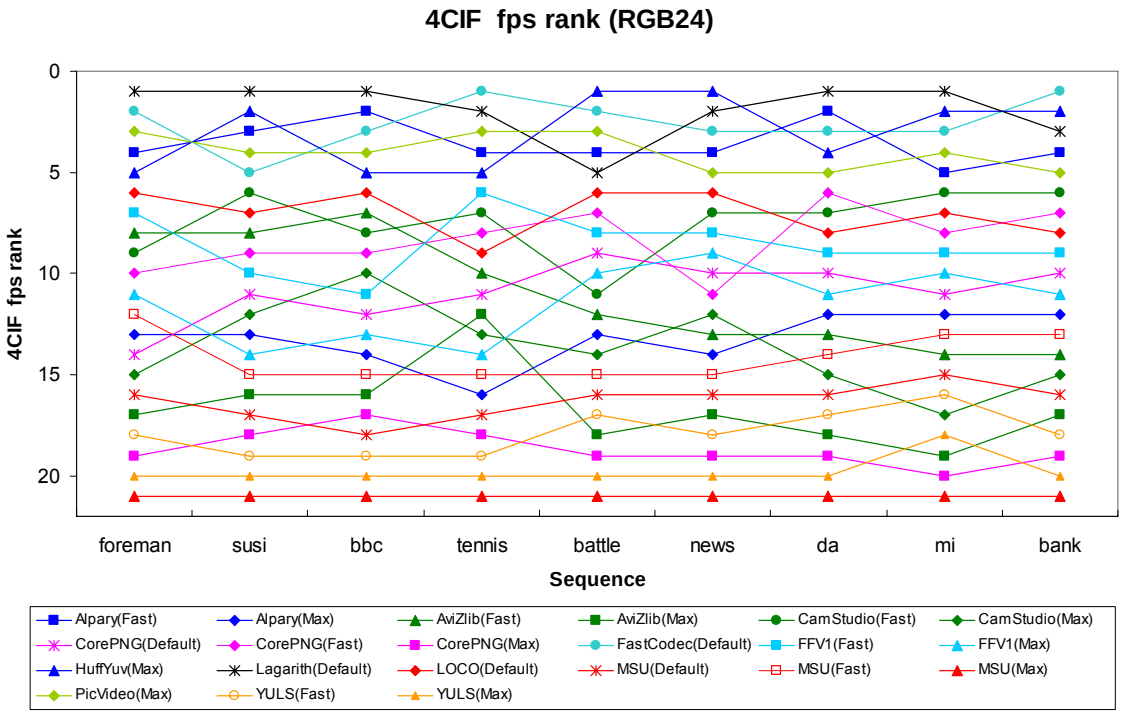


Figure 13. Per-file 4CIF fps ranking for RGB24

As it could be predicted and explained, bigger deviances in normalized encoding speed were noted primarily for dictionary-based codecs like AviZlib. Despite “Battle”, Lagarith shows quite consistent performance being better or on a par with its competitors.

4.2.2 YUY2

The same set of charts as were used for RGB24 color space will be shown and very briefly interpreted.

Figure 14, Table 6, Figure 15 prove that the situation with YUY2 is somewhat different that with RGB24:

- HuffYuv is fastest followed by ArithYuv, Lagarith and Alpary;
- supremacy of YULS by compression ratio is doubtless.

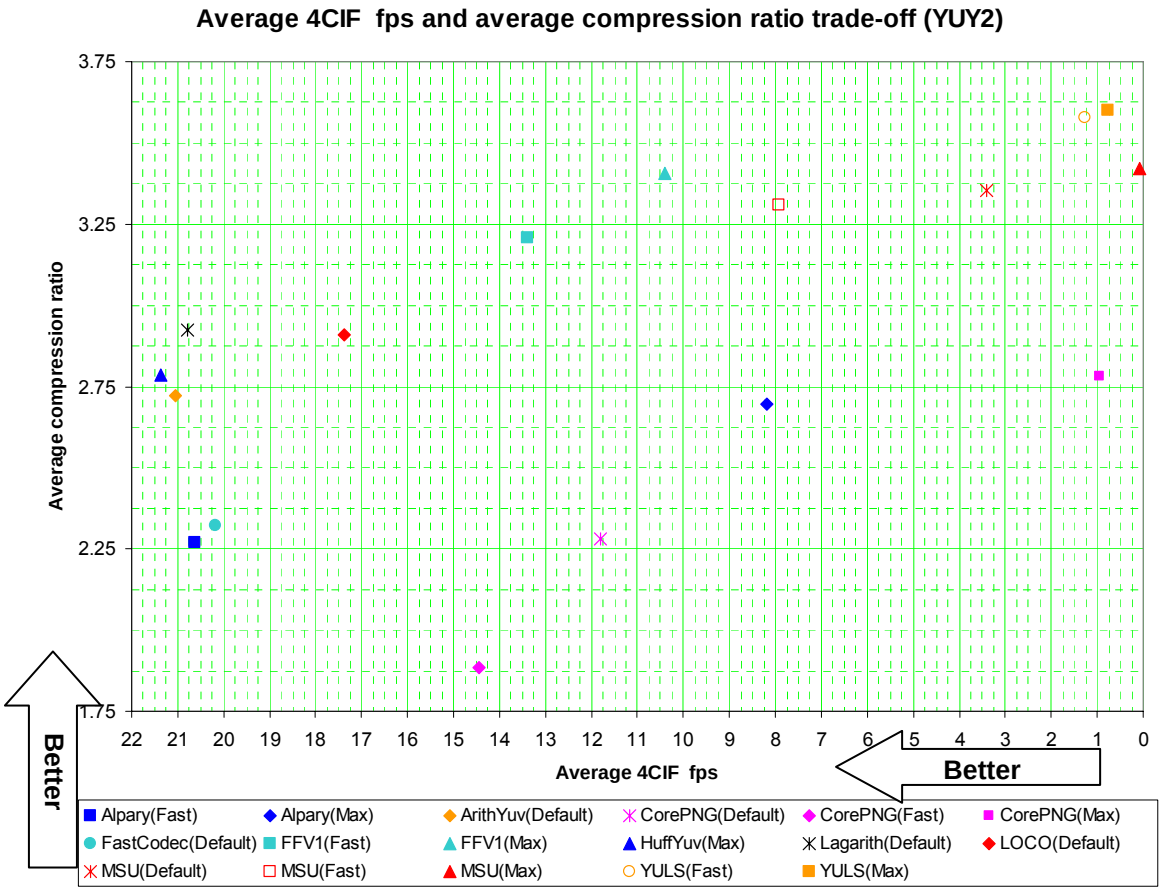


Figure 14. Average 4CIF fps and average compression ratio trade-off for YUY2 color space

Table 6. Average compression ratio and 4CIF fps data for YUY2

Codec	Average compression ratio	Average 4CIF fps
Alpary(Fast)	2,27	20,63
Alpary(Max)	2,70	8,18
ArithYuv(Default)	2,72	21,04
CorePNG(Default)	2,28	11,80
CorePNG(Fast)	1,88	14,45
CorePNG(Max)	2,78	0,94
FastCodec(Default)	2,32	20,18
FFV1(Fast)	3,21	13,40
FFV1(Max)	3,41	10,41
HuffYuv(Max)	2,79	21,37

http://www.compression.ru/video/

Codec	Average compression ratio	Average 4CIF fps
Lagarith(Default)	2,92	20,80
LOCO(Default)	2,91	17,38
MSU(Default)	3,35	3,42
MSU(Fast)	3,31	7,91
MSU(Max)	3,42	0,07
YULS(Fast)	3,58	1,26
YULS(Max)	3,60	0,76

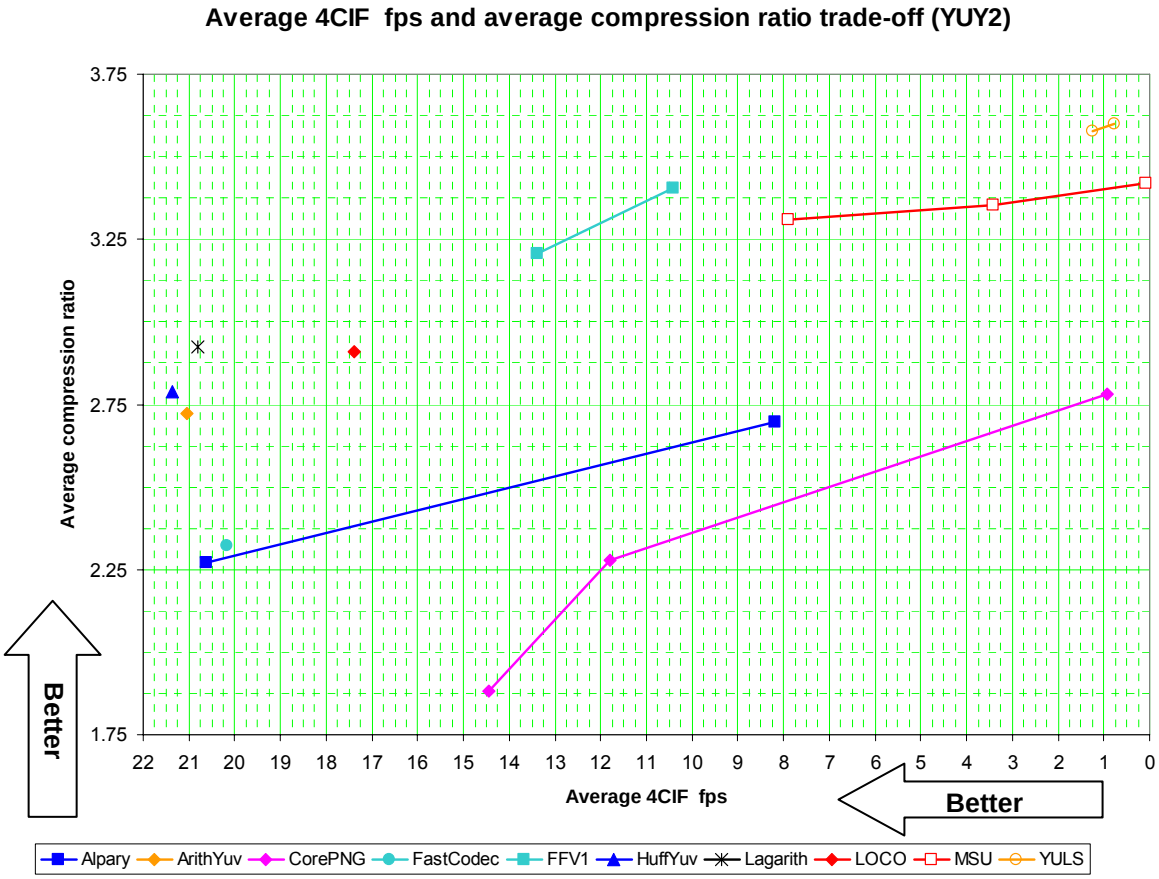


Figure 15. Speed-compression ratio curves for YUY2 color space

It can be seen from the following Figure 16, Figure 17, and Figure 18 that YULS, MSU, and FFV1 makes a group with relatively high compression, YULS is distinctly better. Fastcodec, CorePNG, Alpary provide relatively low compression ratio. Other codecs constitute a group of comparatively medium compression.

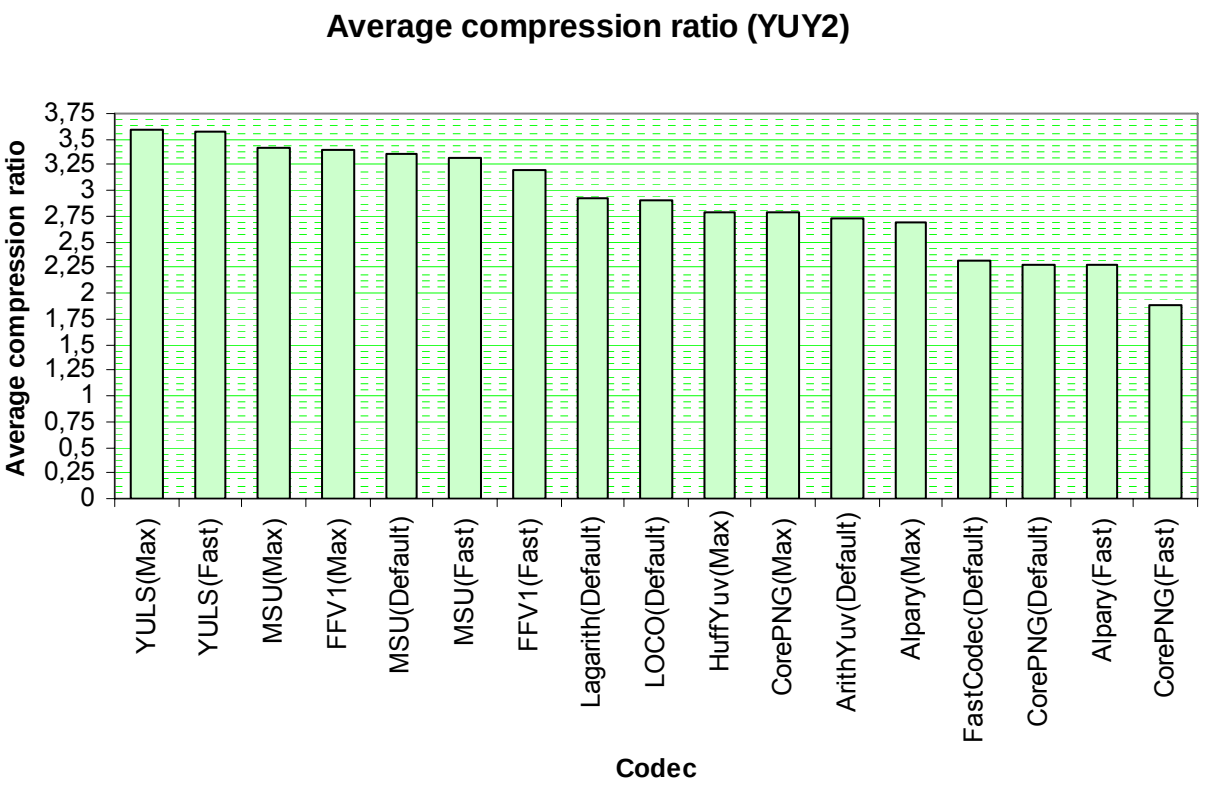


Figure 16. Average compression ratio for YUY2 for codecs and presets

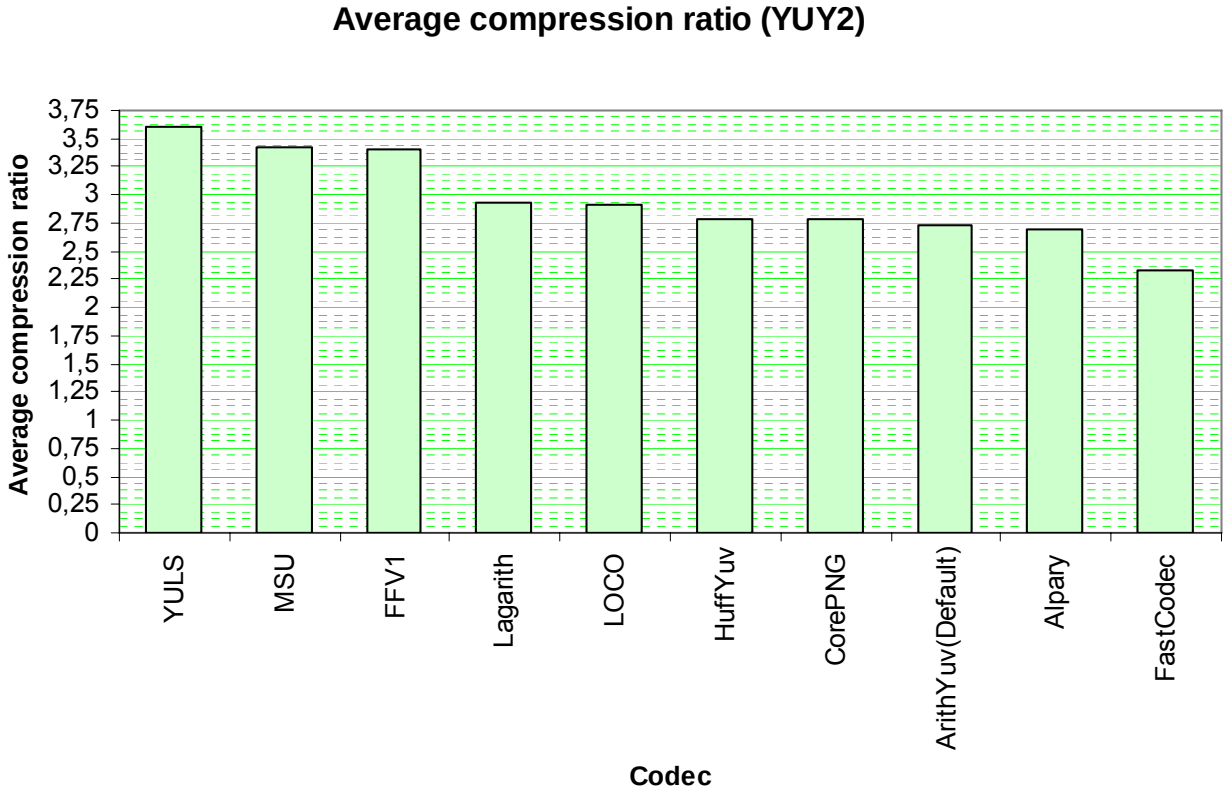


Figure 17. Average compression ratio for YUY2 for codecs only (best results)

http://www.compression.ru/video/

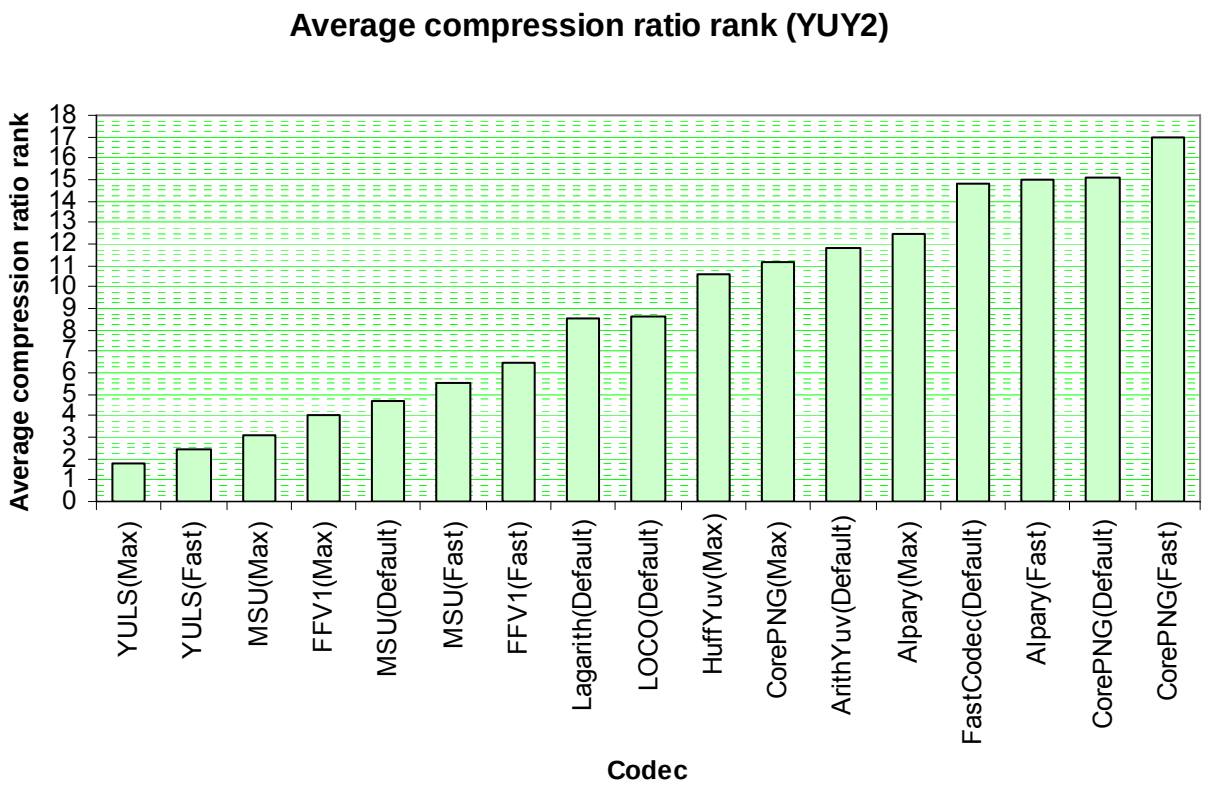


Figure 18. Average compression rank for YUY2 for codecs and presets

Figure 19, Figure 20, Figure 21 highlight leaders by speed. They are HuffYuv, ArithYuv, Lagarith, Alpary and FastCodec.

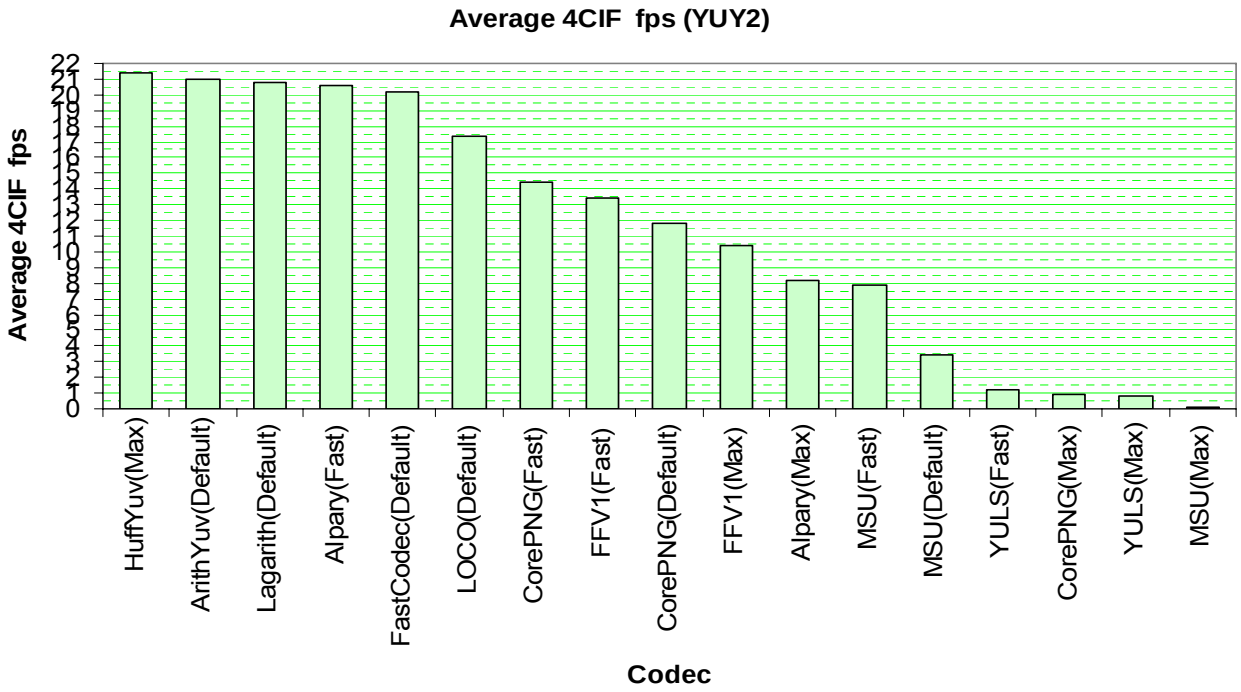


Figure 19. Average 4CIF fps for YUY2 for codecs and presets

http://www.compression.ru/video/

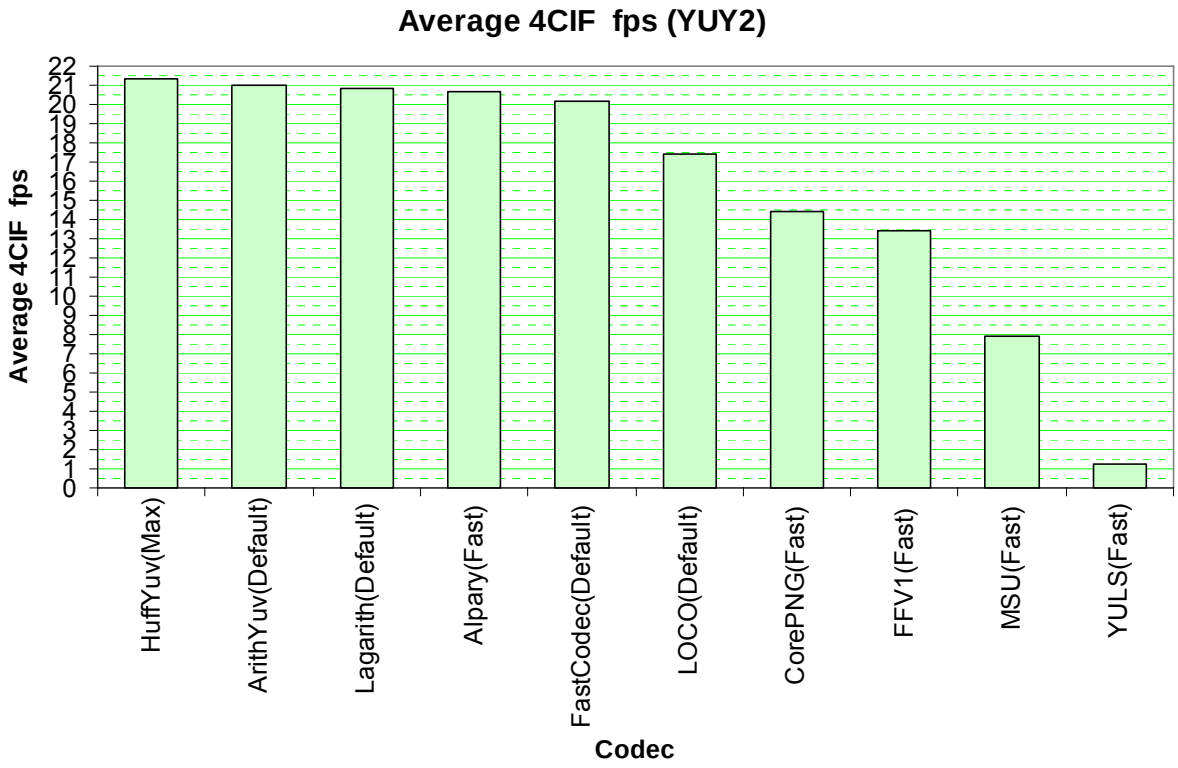


Figure 20. Average 4CIF fps for YUY2 for codecs only (best results)

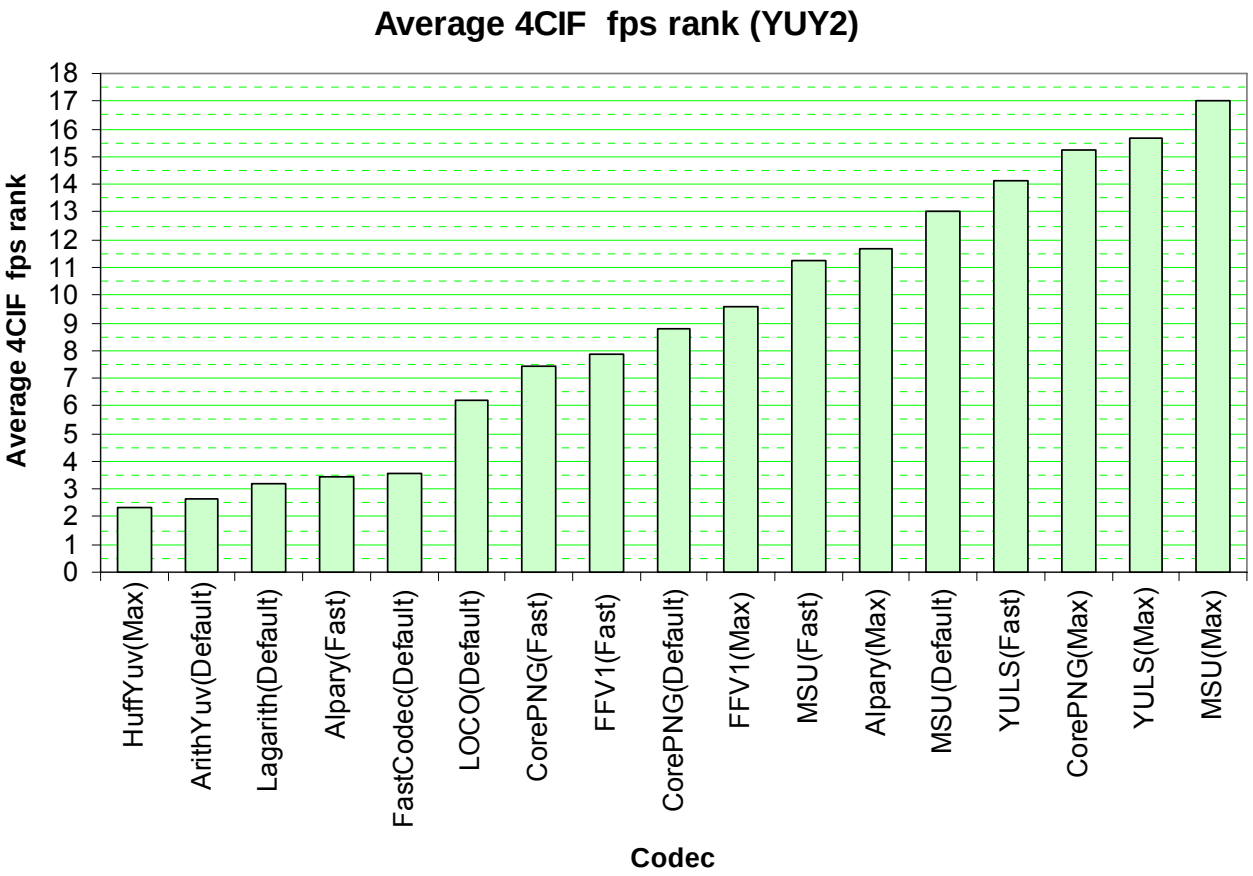


Figure 21. Average 4CIF fps rank for YUY2 for codecs and presets

Per-file results presented on charts below are more consistent and predictable than for RGB24. Generally results correlated those for RGB24. Besides lesser deviations, clear differences are a very good compression ratio of YULS for “News” in comparison to a fall for the corresponding RGB24 video, and big gap in compression ratio between the same codec and its competitors on “Mi” sequence.

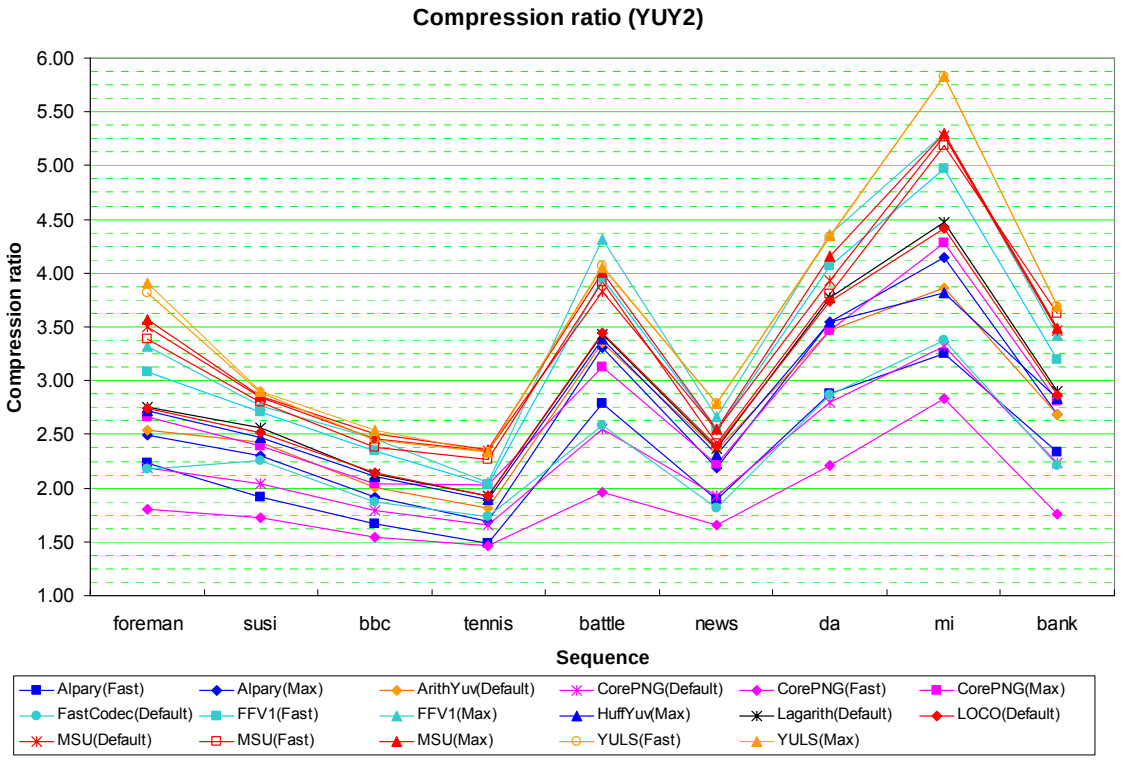


Figure 22. Per-file compression ratio for YUY2

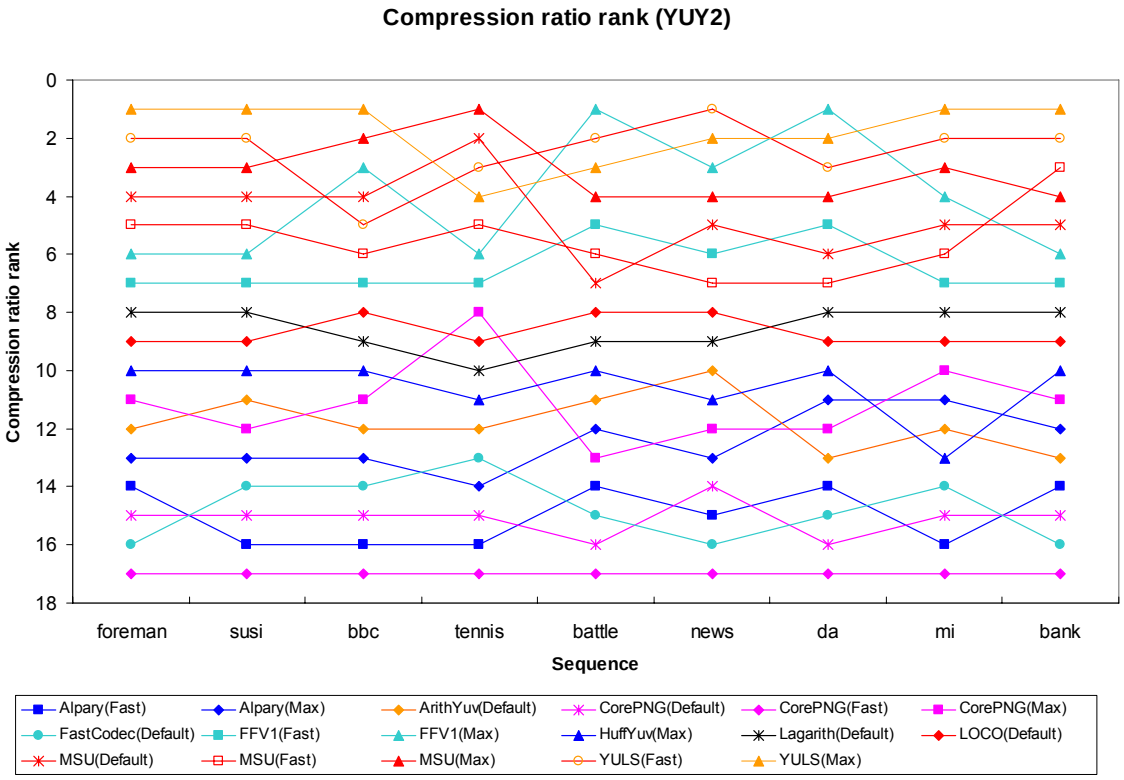


Figure 23. Per-file compression ratio ranking for YUY2

http://www.compression.ru/video/

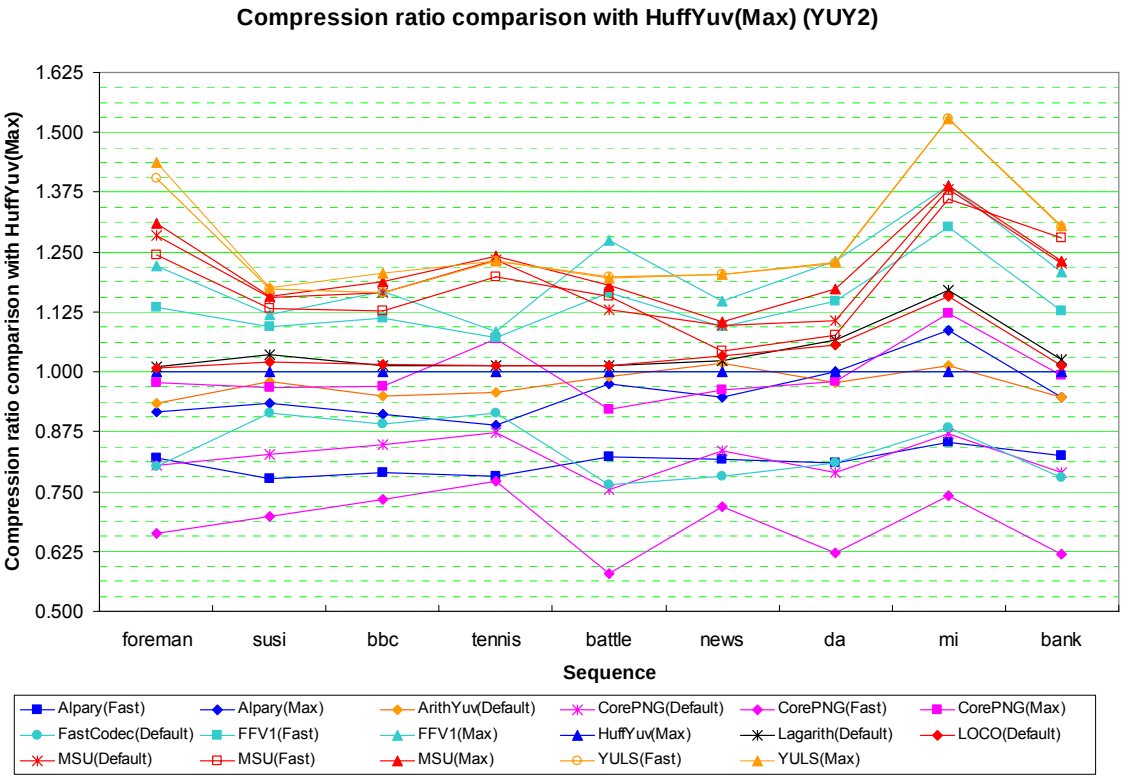


Figure 24. Relative per-file compression ratio for YUY2 in comparison to HuffYuv

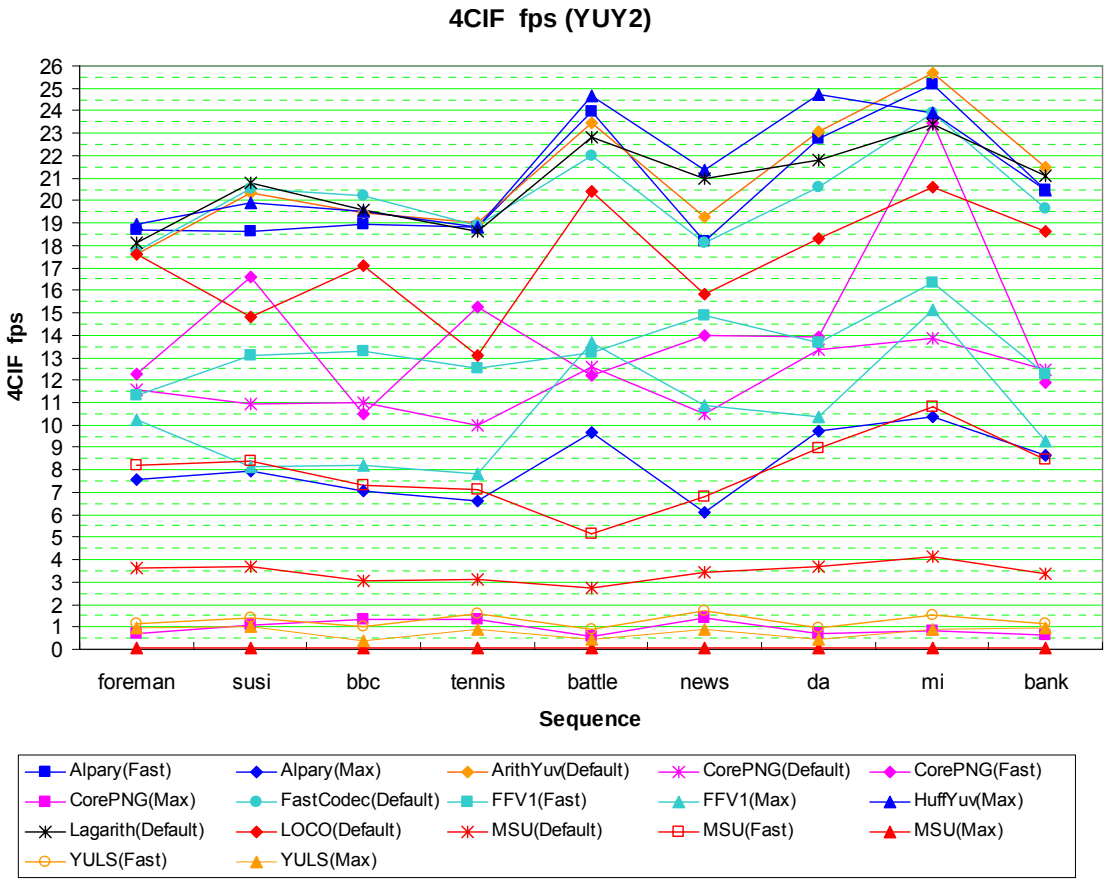


Figure 25. Per-file 4CIF fps for YUY2

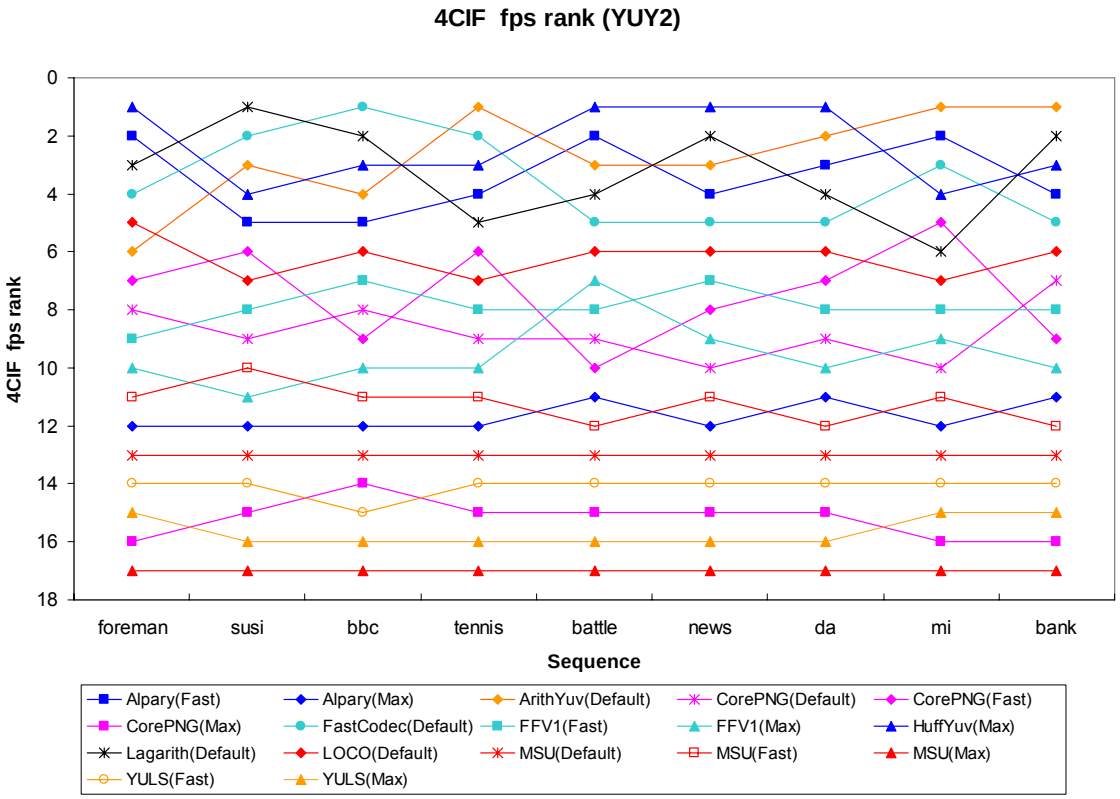


Figure 26. Per-file 4CIF fps ranking for YUY2

Probably, among not the slowest presets, the most predictable speed is shown by FFV1 “Fast” and Loco “Default”.

4.2.3 YV12

The same set of charts as were used for RGB24 and YUY2 color spaces will be shown here and very briefly interpreted.

The general situation for YV12 color spaces is similar to YUY2. It can be characterized as follows:

- HuffYuv is fastest;
- HuffYuv and Lagarith show better speed/compression trade-off in a group of relatively fast codecs;
- YULS provides clearly the best compression.

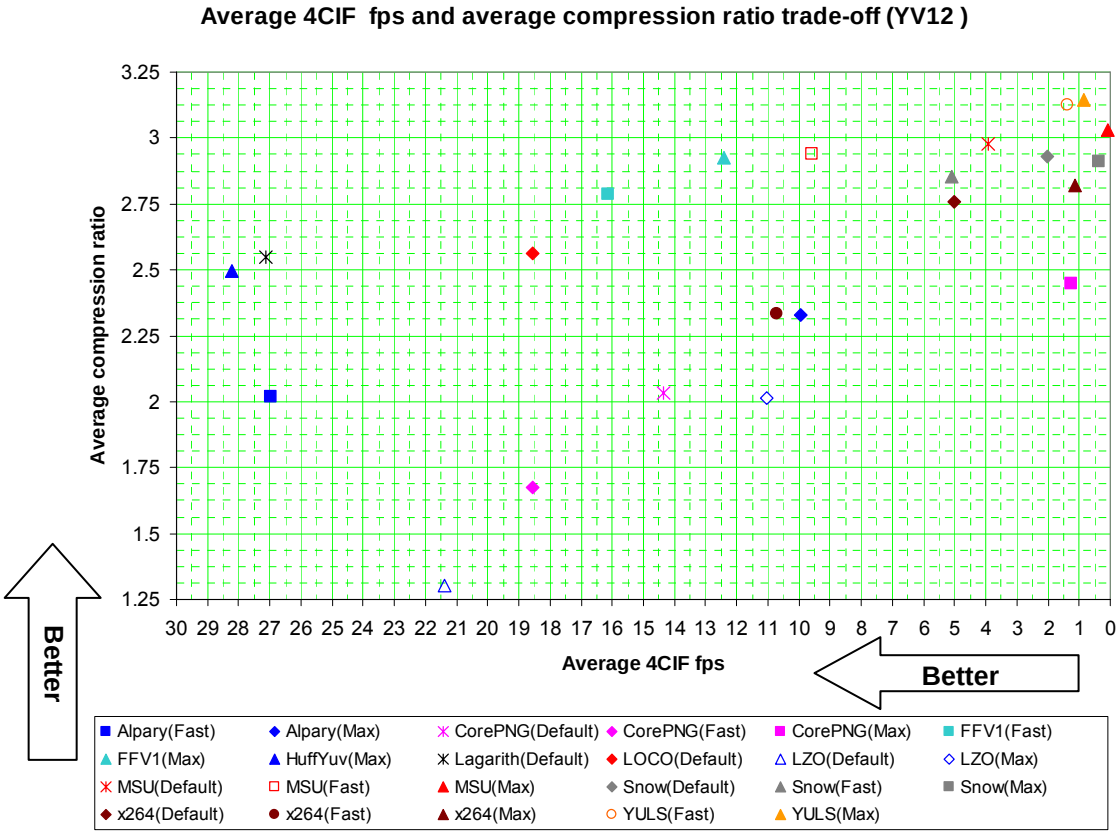


Figure 27. Average 4CIF fps and average compression ratio trade-off for YV12

Table 7. Average compression ratio and 4CIF fps data for YV12

Codec	Average compression ratio	Average 4CIF fps
Alpary(Fast)	2,02	26,98
Alpary(Max)	2,33	9,95
CorePNG(Default)	2,03	14,35
CorePNG(Fast)	1,67	18,57
CorePNG(Max)	2,45	1,26
FFV1(Fast)	2,78	16,12
FFV1(Max)	2,92	12,41
HuffYuv(Max)	2,49	28,24
Lagarith(Default)	2,55	27,13
LOCO(Default)	2,56	18,56
LZO(Default)	1,30	21,39
LZO(Max)	2,01	11,04
MSU(Default)	2,98	3,92
MSU(Fast)	2,94	9,60
MSU(Max)	3,03	0,07
Snow(Default)	2,93	2,00
Snow(Fast)	2,85	5,11
Snow(Max)	2,91	0,37
x264(Default)	2,76	5,01
x264(Fast)	2,33	10,73
x264(Max)	2,82	1,15
YULS(Fast)	3,13	1,36
YULS(Max)	3,14	0,84

Average 4CIF fps and average compression ratio trade-off (YV12)

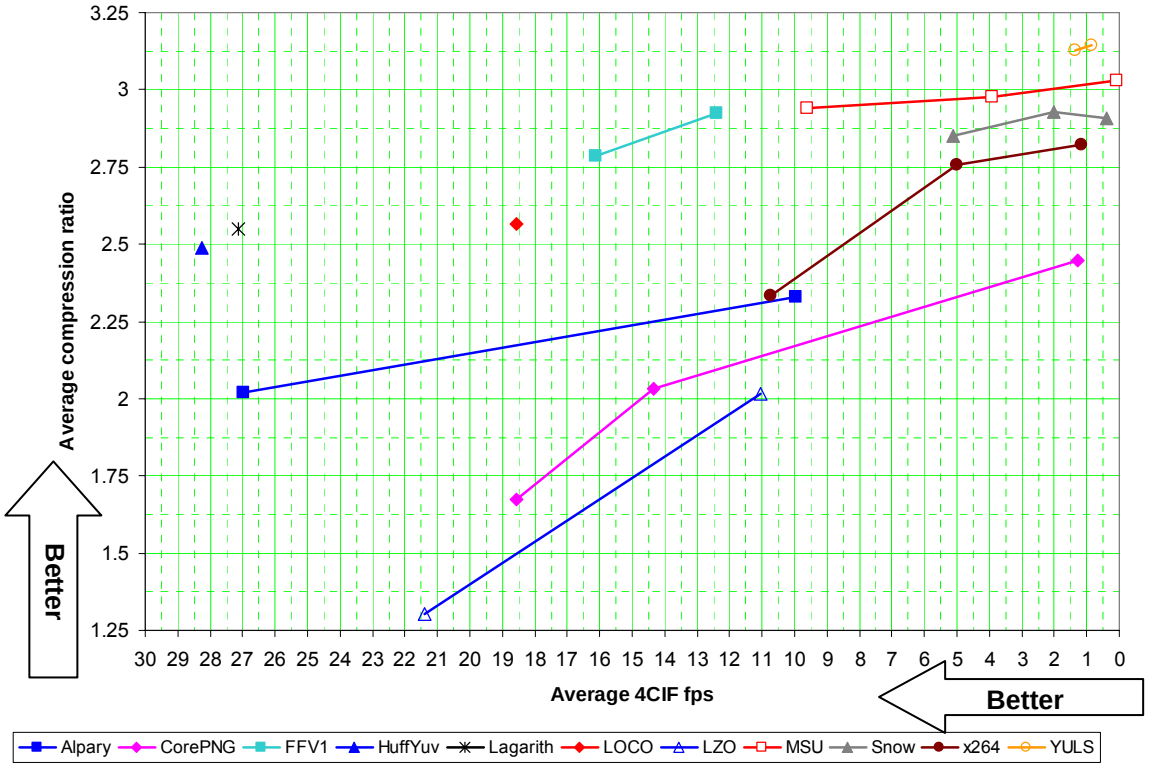


Figure 28. Speed-compression ratio curves for YV12

http://www.compression.ru/video/

Looking at Figure 28, Figure 29, and Figure 32 it should be noted that x264 lossless compression mode tested in this part of comparison is clearly beaten by FFV1, MSU, and Snow both by speed and compression for our test set. LZO has the same problem in its niche.

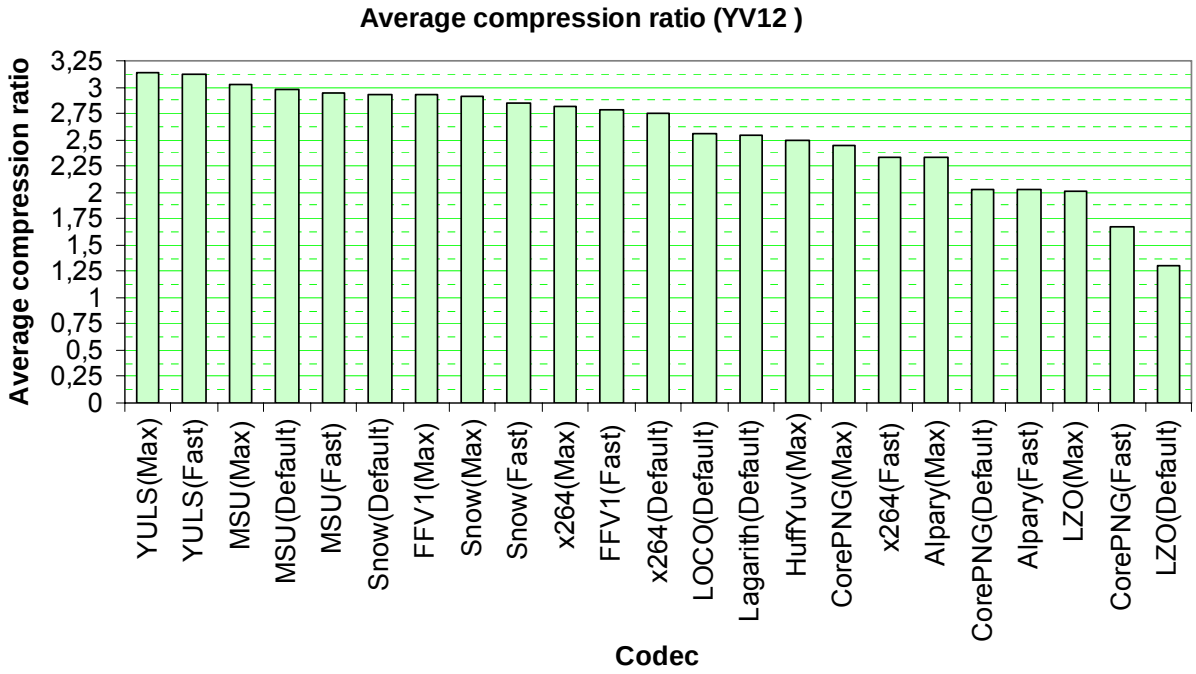


Figure 29. Average compression ratio for YV12 for codecs and presets

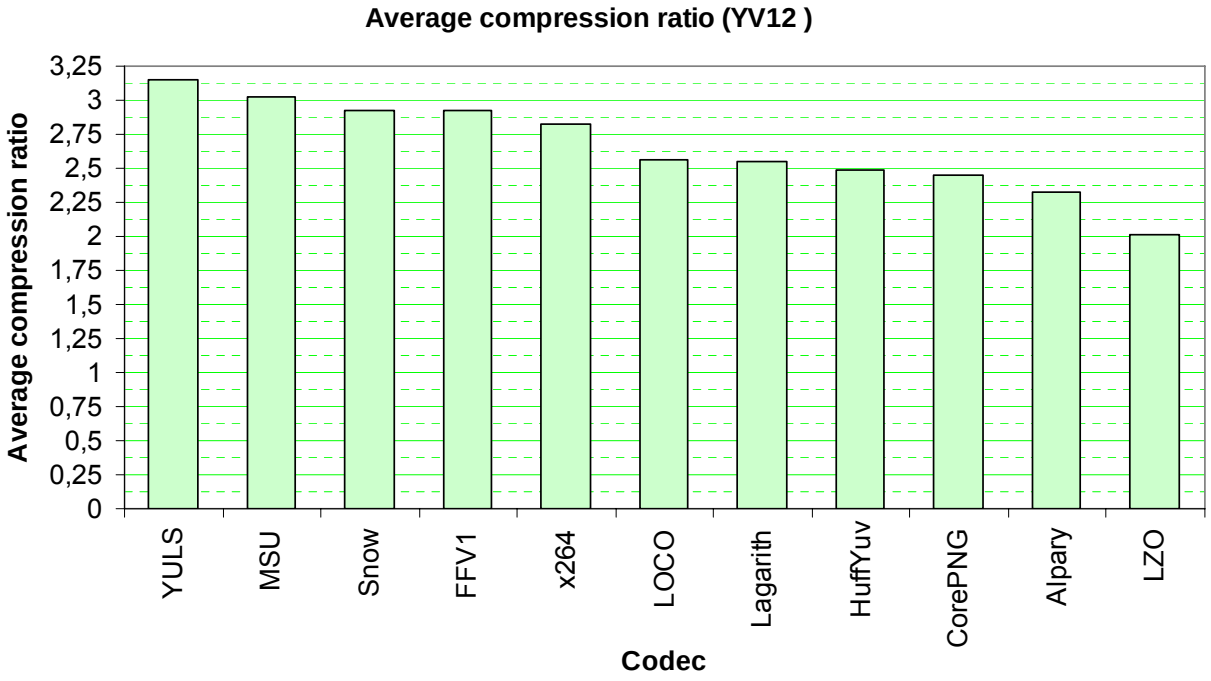


Figure 30. Average compression ratio for YV12 for codecs only (best results)

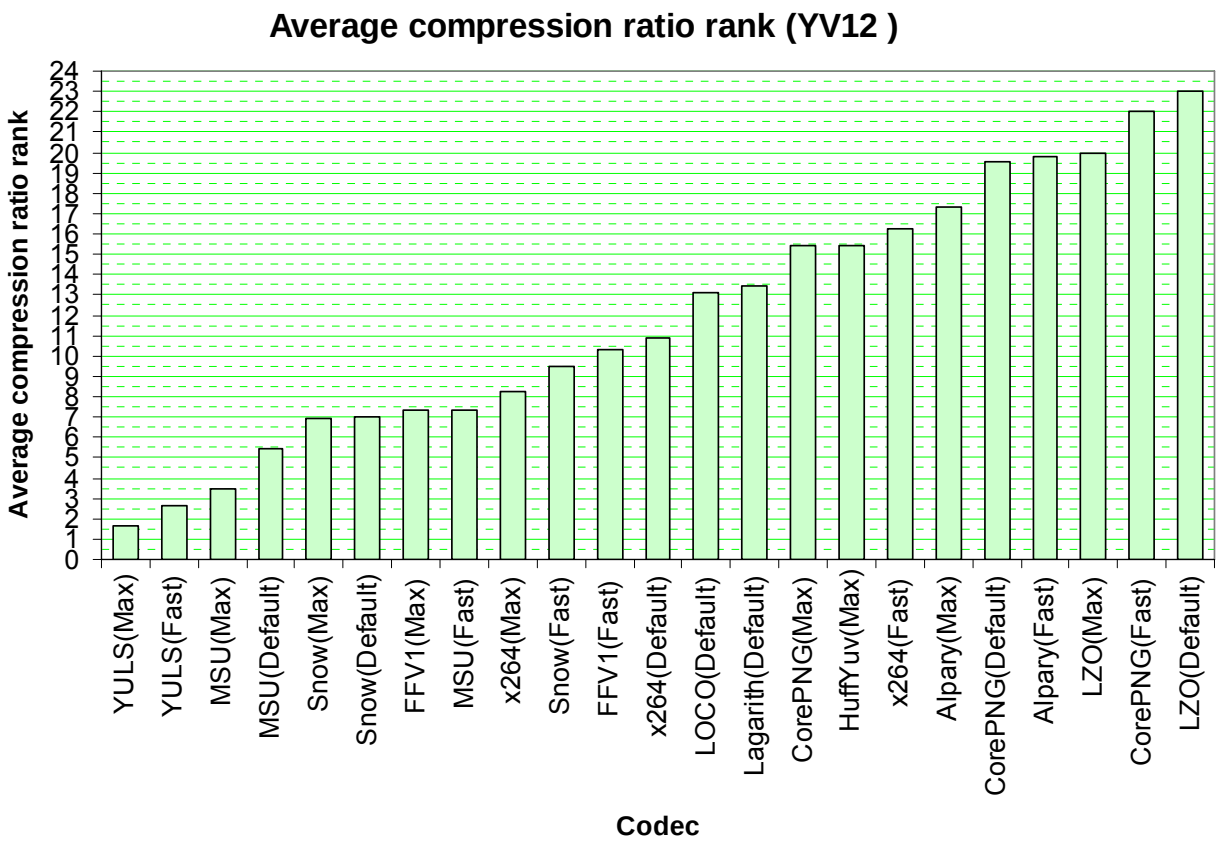


Figure 31. Average compression rank for YV12 for codecs and presets

The slowest codecs and presets are, as before, MSU “Max”, YULS, CorePNG “Max” with an addition of Snow and x264 “Max”.

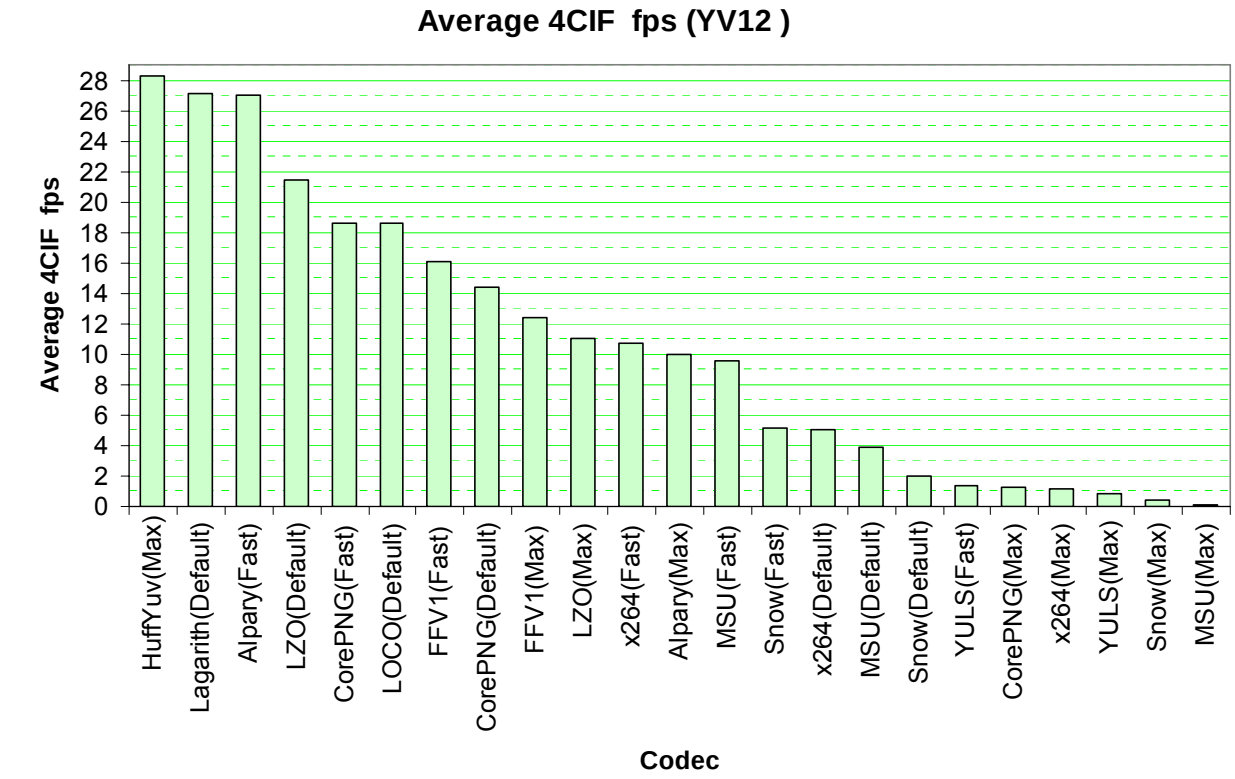


Figure 32. Average 4CIF fps for YV12 for codecs and presets

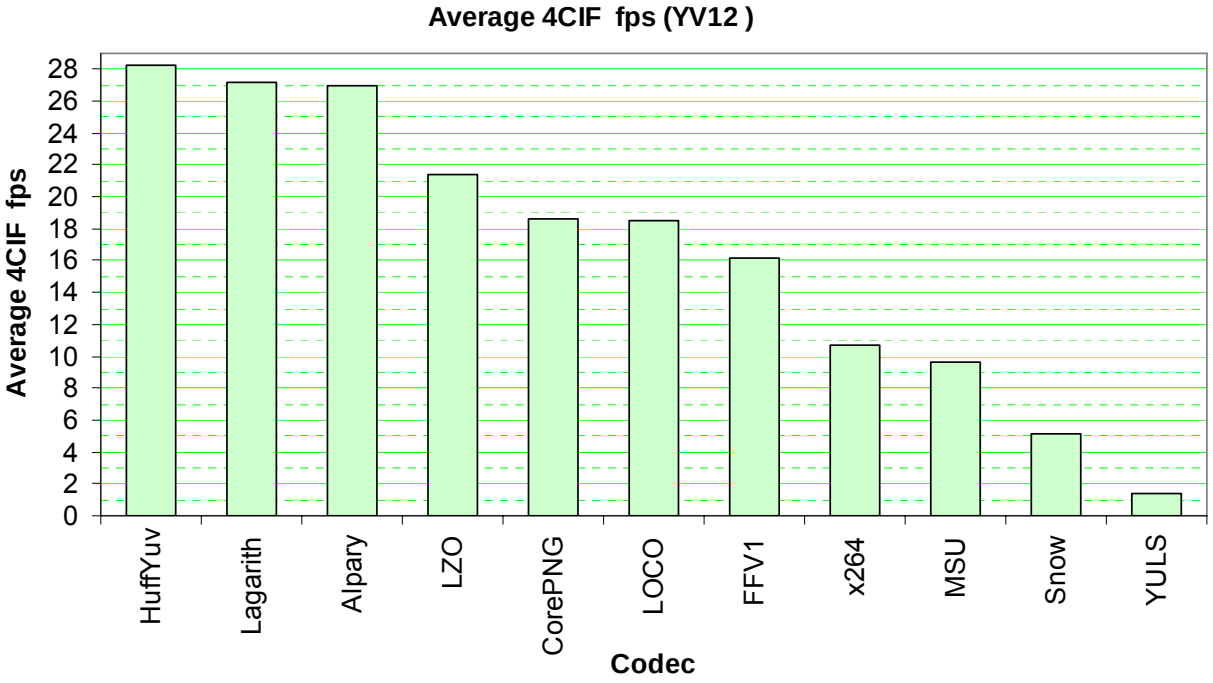


Figure 33. Average 4CIF fps for YV12 for codecs only (best results)

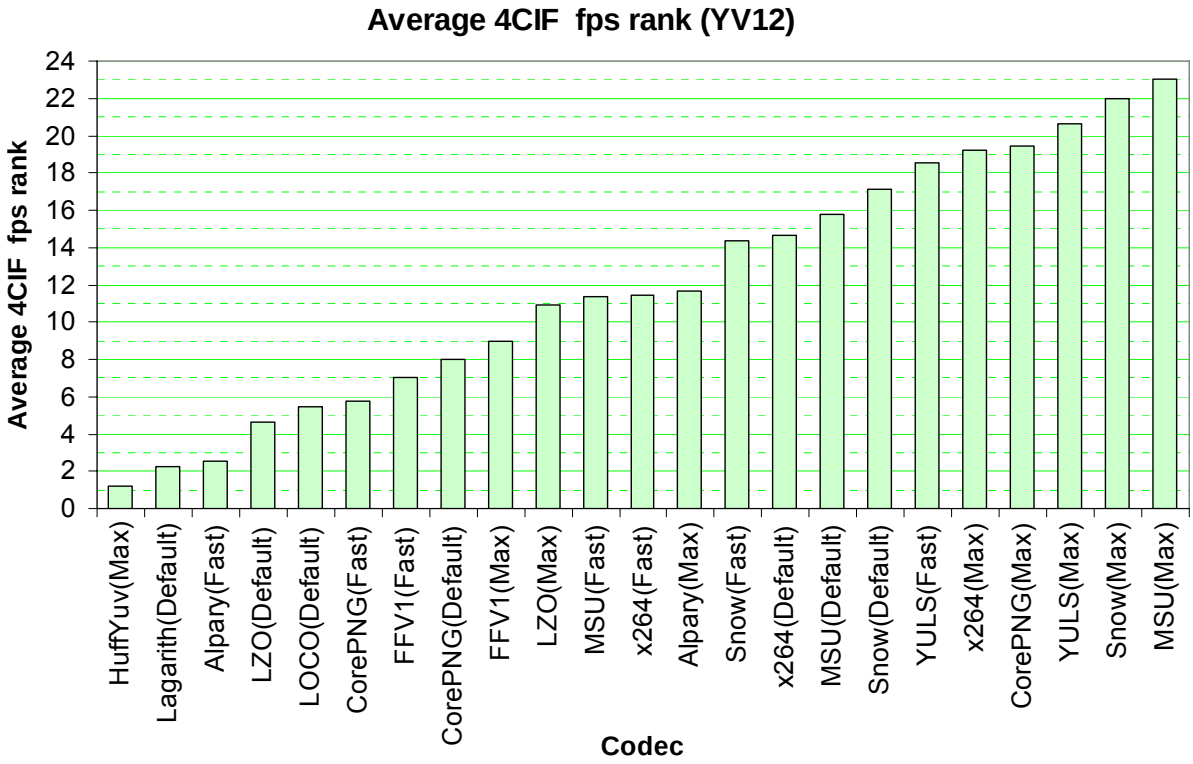


Figure 34. Average 4CIF fps rank for YV12 for codecs and presets

Detailed results for each file presented on the following charts Figure 35 - Figure 39 are similar to those of YUY2. The features are:

- there is a clear outsider by compression – LZO;
- advantage of YULS by compression ratio on “Foreman” is not so big comparatively;
- relative compression ratio of FFV1 “Max” is much less predictable.

It should also be mentioned that compression ranking of x264 has big deviation among files and sometimes lower than that of HuffYuv and with considerably worse encoding speed.

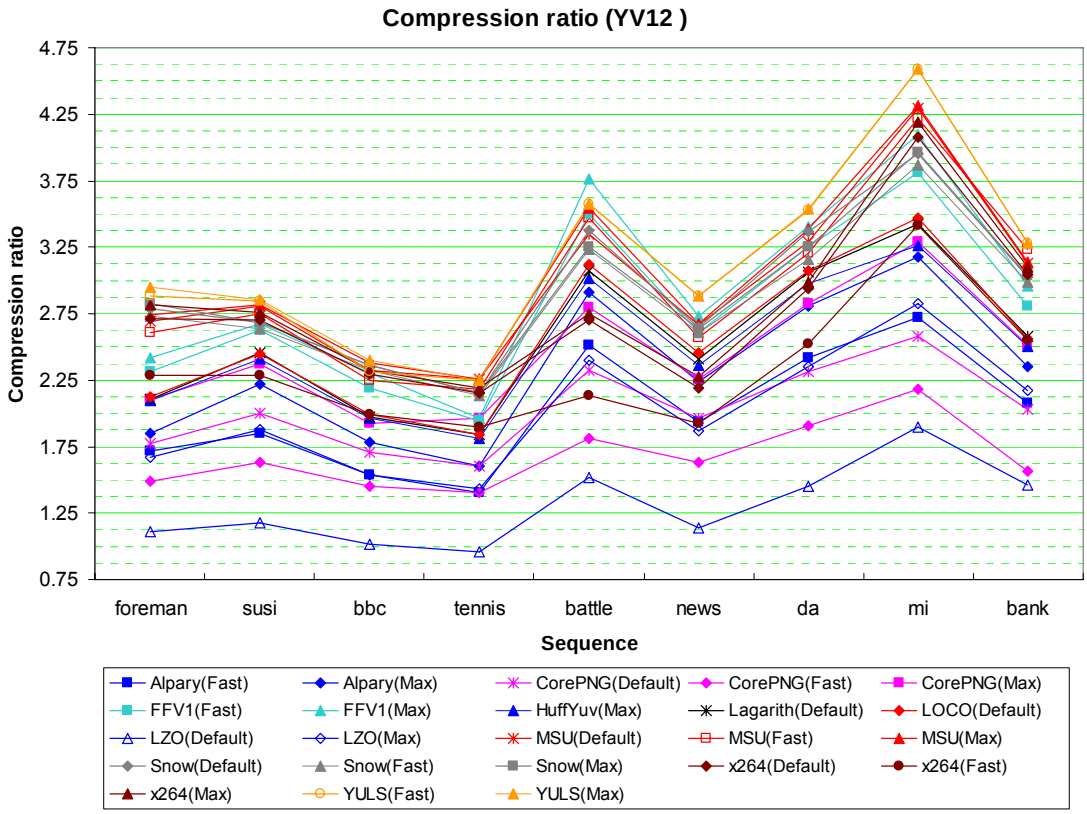


Figure 35. Per-file compression ratio for YV12

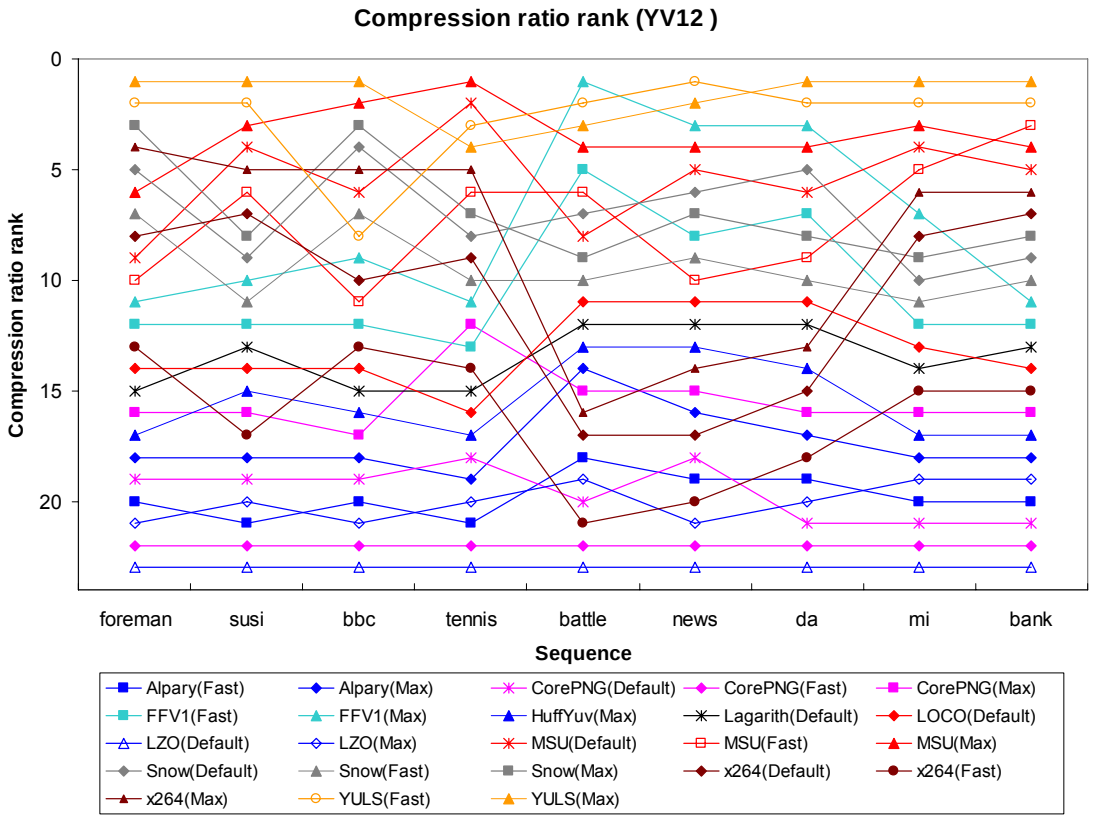


Figure 36. Per-file compression ratio ranking for YV12

http://www.compression.ru/video/

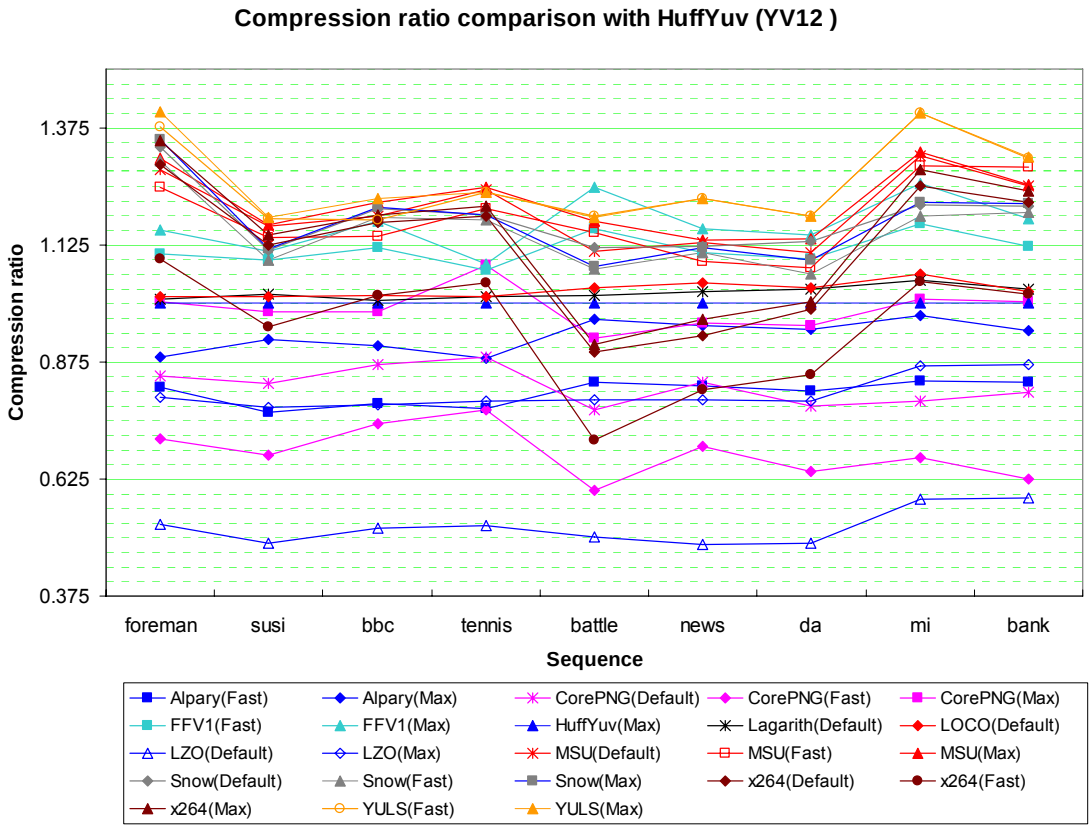


Figure 37. Relative per-file compression ratio for YV12 in comparison to HuffYuv

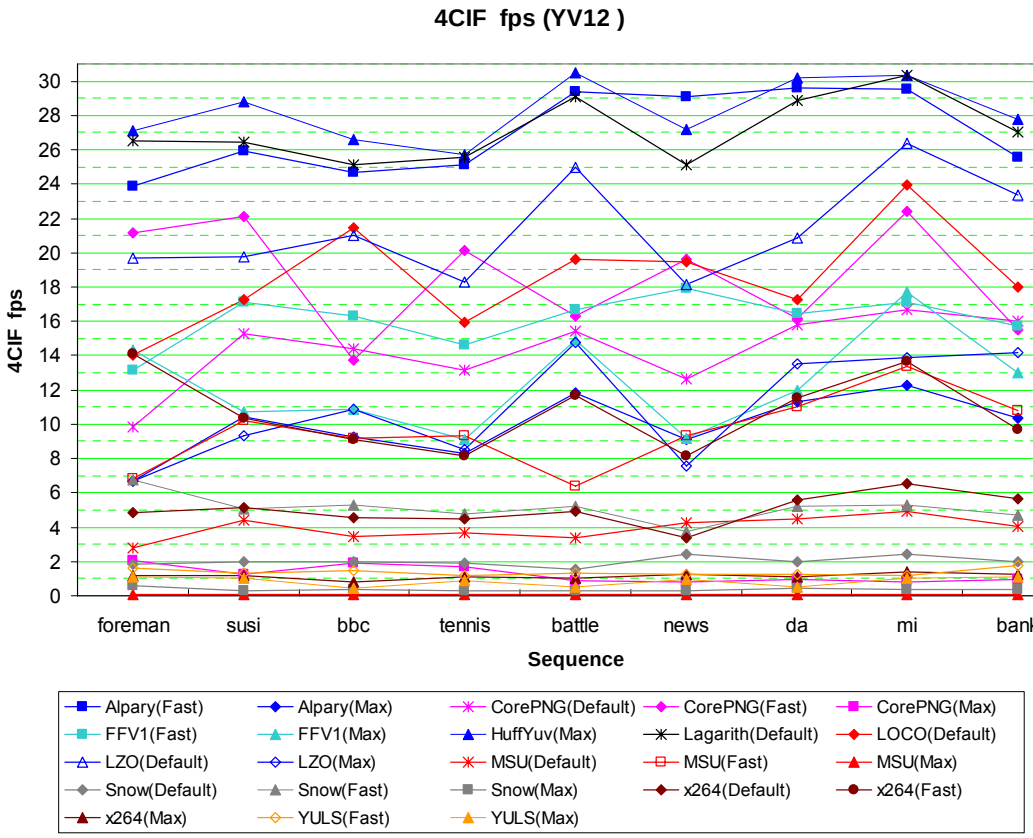


Figure 38. Per-file 4CIF fps for YV12

http://www.compression.ru/video/

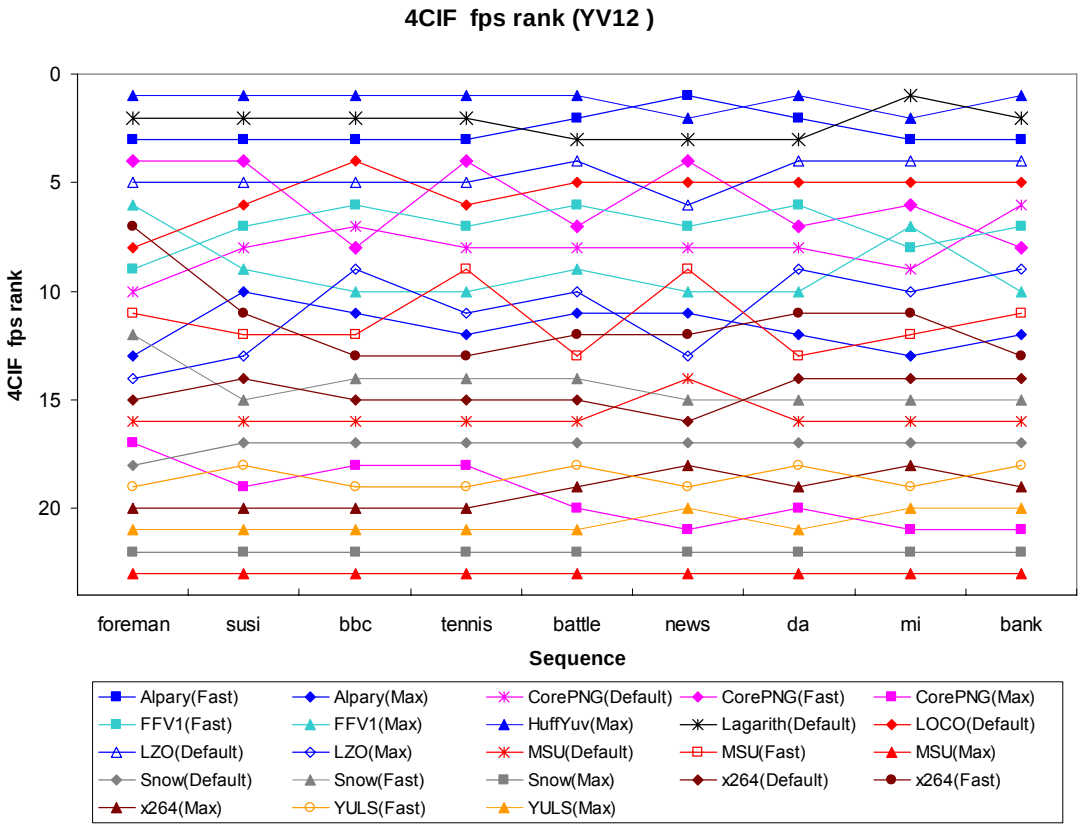


Figure 39. Per-file 4CIF fps ranking for YV12

4.2.4 Analysis and Choice of Application Area

For further analysis it is reasonable to divide codecs into two specific groups, which can be named:

- Video Capture and Video Editing Area
- Maximum Compression Area

In YUY2 and YV12 the Video Capture and Video Editing Area group was chosen to consist of codecs with average 4CIF fps being more than 20. In case of RGB24 (where all average 4CIF fps are smaller than 20) it includes first five best by encoding speed codecs with small differences in speed between each other. The list of codecs with presets included into Video Capture and Video Editing Area group is as follows:

RGB24

- Alpary (Fast)
- FastCodec (Default)
- HuffYuv (Max)
- Lagarith (Default)
- PICVideo (Fast)

YUY2

- Alpary (Fast)
- ArithYuv (Default)
- FastCodec (Default)
- HuffYuv (Max)
- Lagarith (Default)

YV12

- Alpary (Fast)
- HuffYuv (Max)
- Lagarith (Default)
- LZO (Default)

Maximum Compression Area consists of codecs with average compression ratio greater or equal to compression ratio of HuffYuv “Max” in at least one color space. The list of codecs with presets in Maximum Compression Area group is:

RGB24

- Alpary (Fast)
- Alpary (Max)
- FastCodec (Default)
- FFV1 (Max)
- FFV1 (Fast)
- HuffYuv (Max)
- Lagarith (Default)
- MSU (Max)
- MSU (Default)
- MSU (Fast)
- YULS (Max)
- YULS (Fast)

YUY2

- FFV1 (Max)
- FFV1 (Fast)
- HuffYuv (Max)
- Lagarith (Default)
- LOCO (Default)
- MSU (Max)
- MSU (Default)
- MSU (Fast)
- YULS (Max)
- YULS (Fast)

YV12

- FFV1 (Max)
- FFV1 (Fast)
- HuffYuv (Max)
- Lagarith (Default)
- LOCO (Default)
- MSU (Max)
- MSU (Default)
- MSU (Fast)
- Snow (Max)
- Snow (Default)
- Snow (Fast)
- YULS (Max)
- YULS (Fast)
- x264 (Max)
- x264 (Default)

4.3 Video Capture and Video Editing Area

4.3.1 RGB24

As it is demonstrated on Figure 40, the convincing **leader** in RGB24 color space **is Lagarith**. It provides the best result both by compression ratio and speed for our test set and our testing hardware with approximately **30% better compression** and **5-8% better speed**. **Lagarith is recommended to use in any situations.**

The rest codecs show basically similar results, FastCodec and HuffYuv being slightly better. Differences between FastCodec and HuffYuv, as well as between Alpary and PicVideo, are probably within an error of measurements (testing methodology), and should not be trusted to make any distinctive conclusion on their mutual standing.

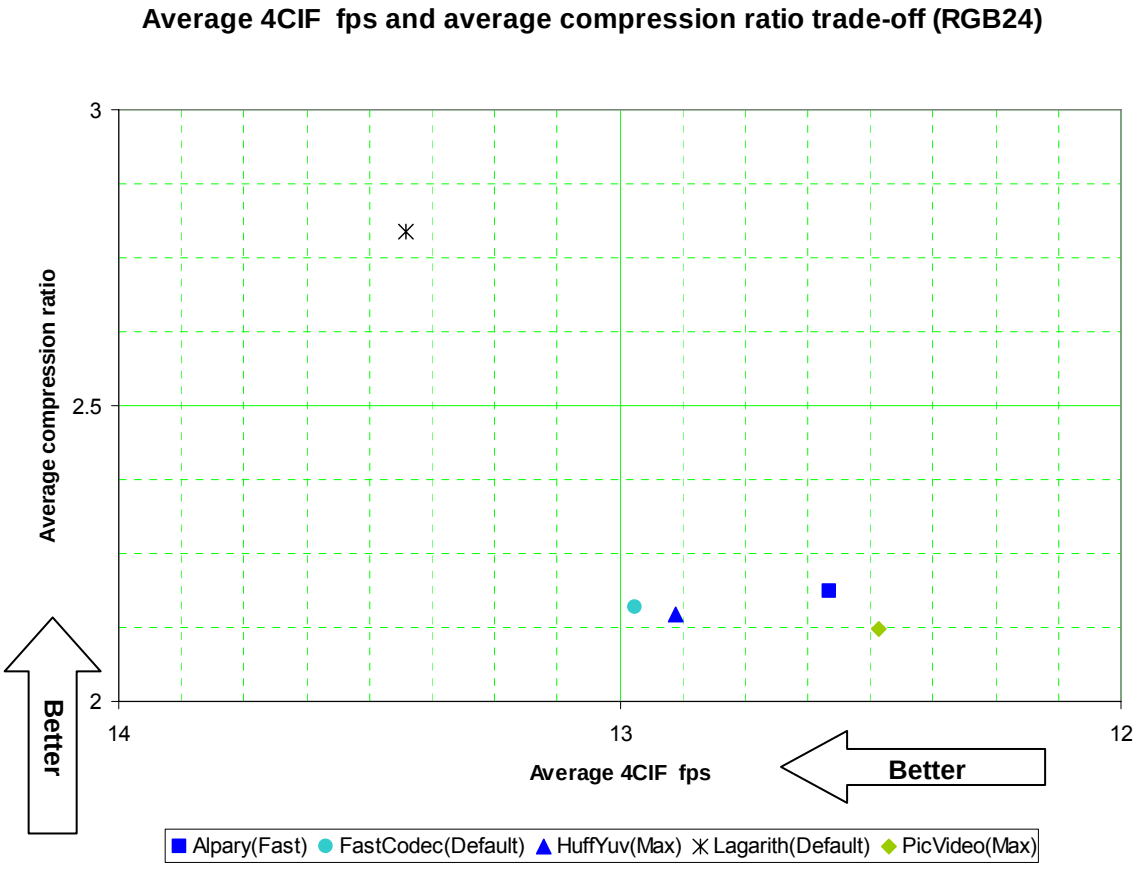


Figure 40. Video Capture: Average 4CIF fps and average compression ratio trade-off for RGB24

The next charts stress results from the standpoint of a certain criterion of comparison and provide details on per-file results.

Figure 41 and Figure 42 underline compression ratio results only. As it can be seen from Figure 42, Lagarith very consistently provides the best compression.

http://www.compression.ru/video/

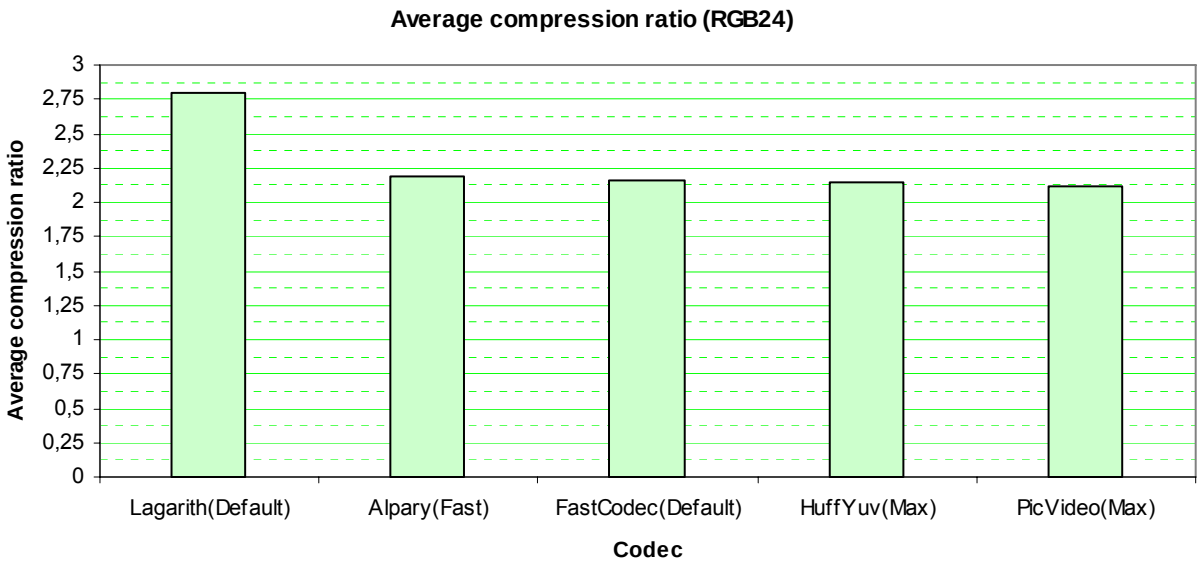


Figure 41. Video Capture: Average compression ratio for RGB24

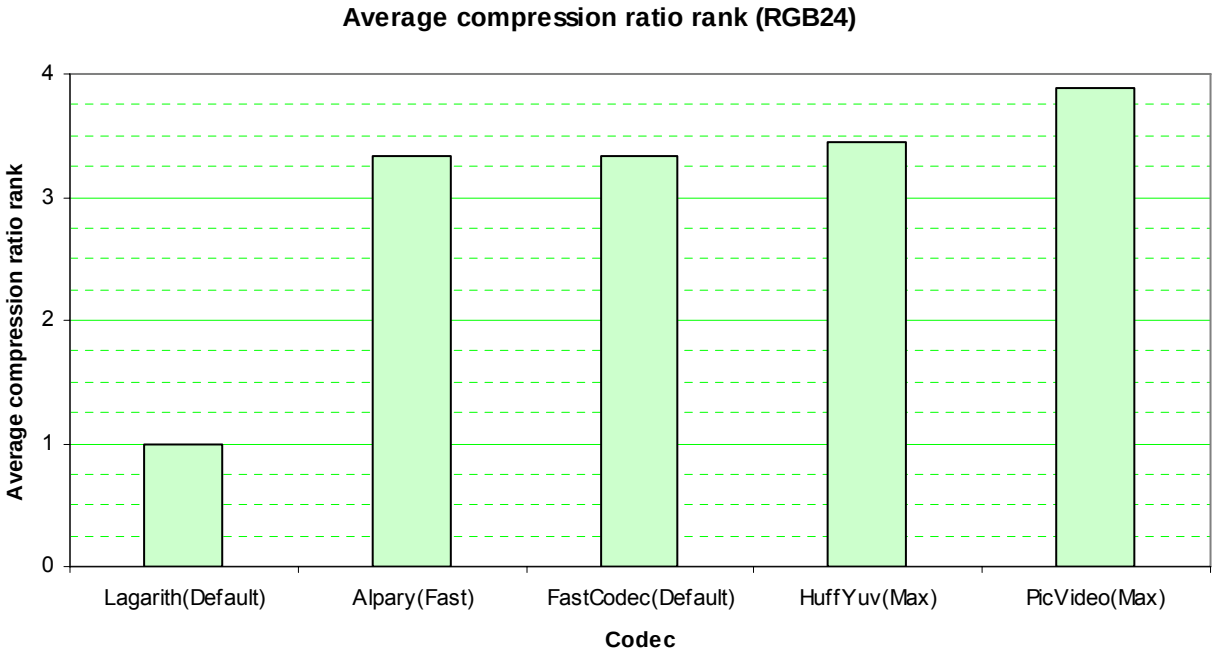


Figure 42. Video Capture: Average compression rank for RGB24

Speed only results are visualized on Figure 43 and Figure 44.

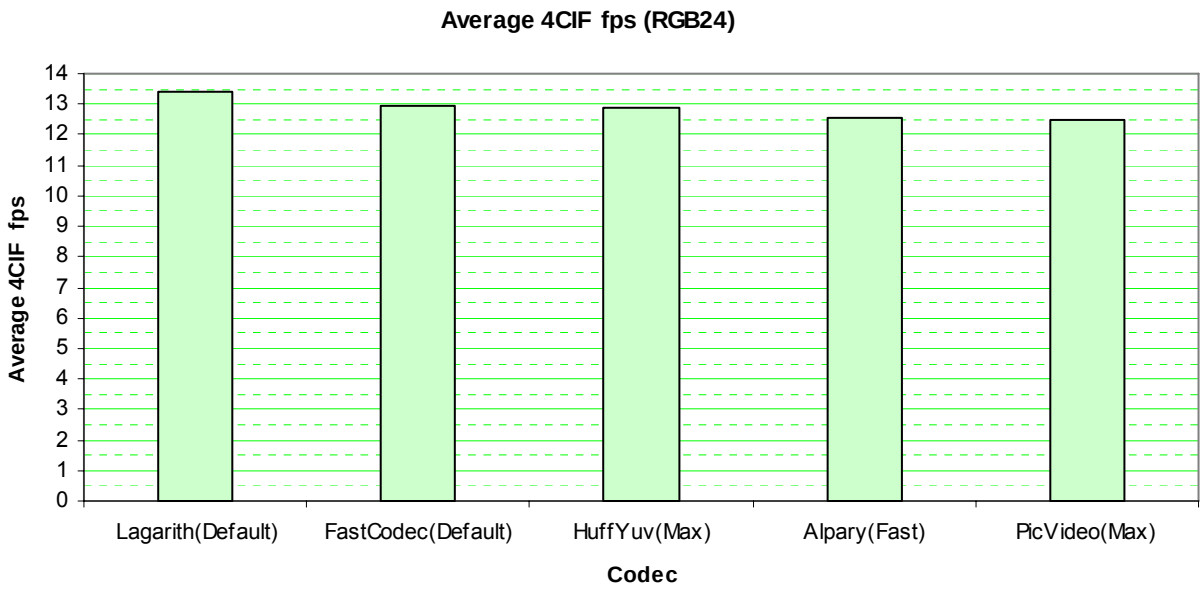


Figure 43. Video Capture: Average 4CIF fps for RGB24

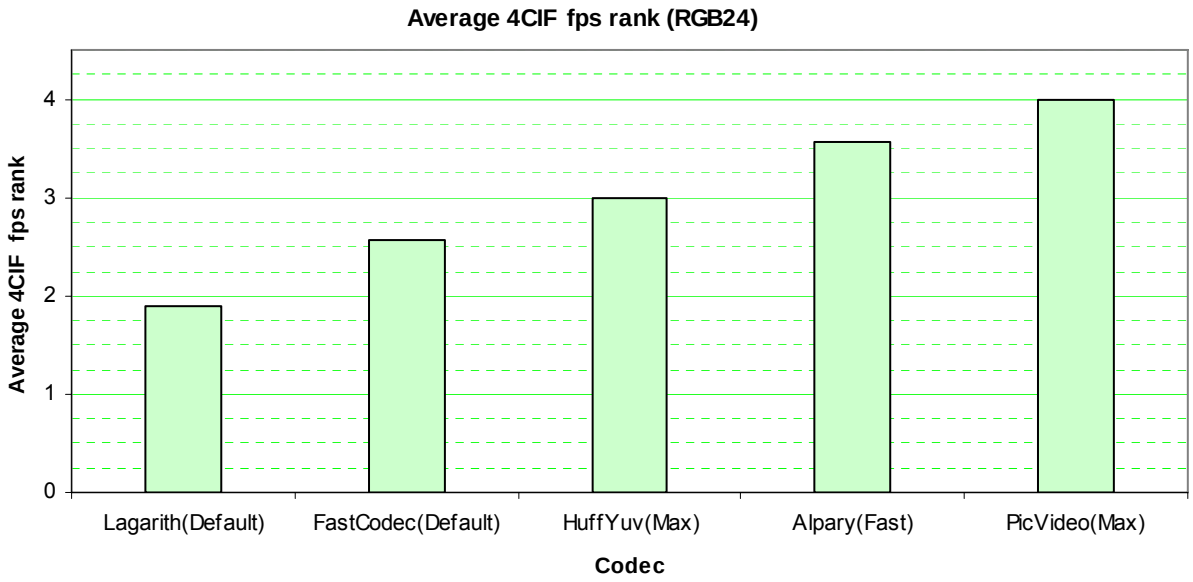


Figure 44. Video Capture: Average 4CIF fps rank for RGB24

Per-file compression ratio results on Figure 45, Figure 46, Figure 47 show some information about Alpary, FastCodec and PicVideo behaviour on different video. It can be seen that deviation of results may be quite big for certain files, about ±20% for our test set, while average compression ratios are very close to each other.

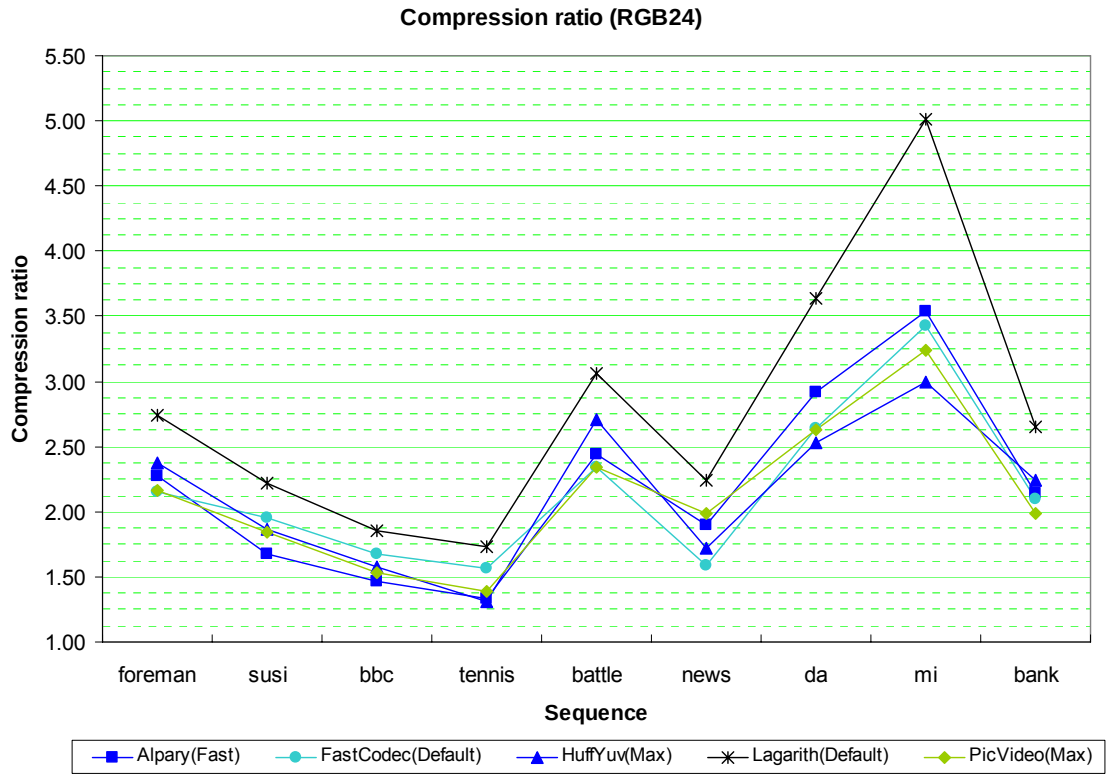


Figure 45. Video Capture: Compression ratio for each sequence in RGB24

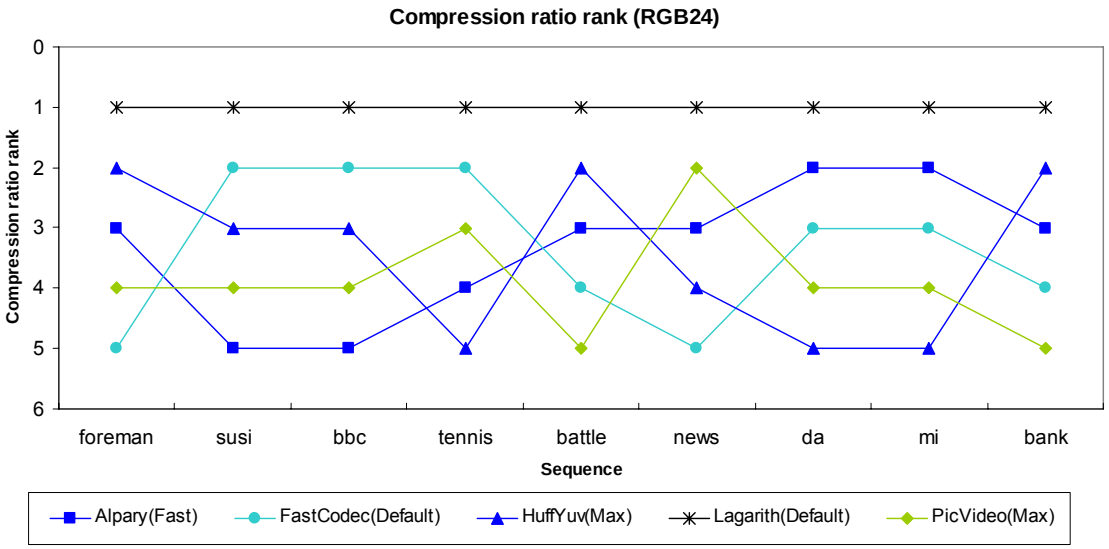


Figure 46. Video Capture: Per-file compression ratio ranking for RGB24

http://www.compression.ru/video/

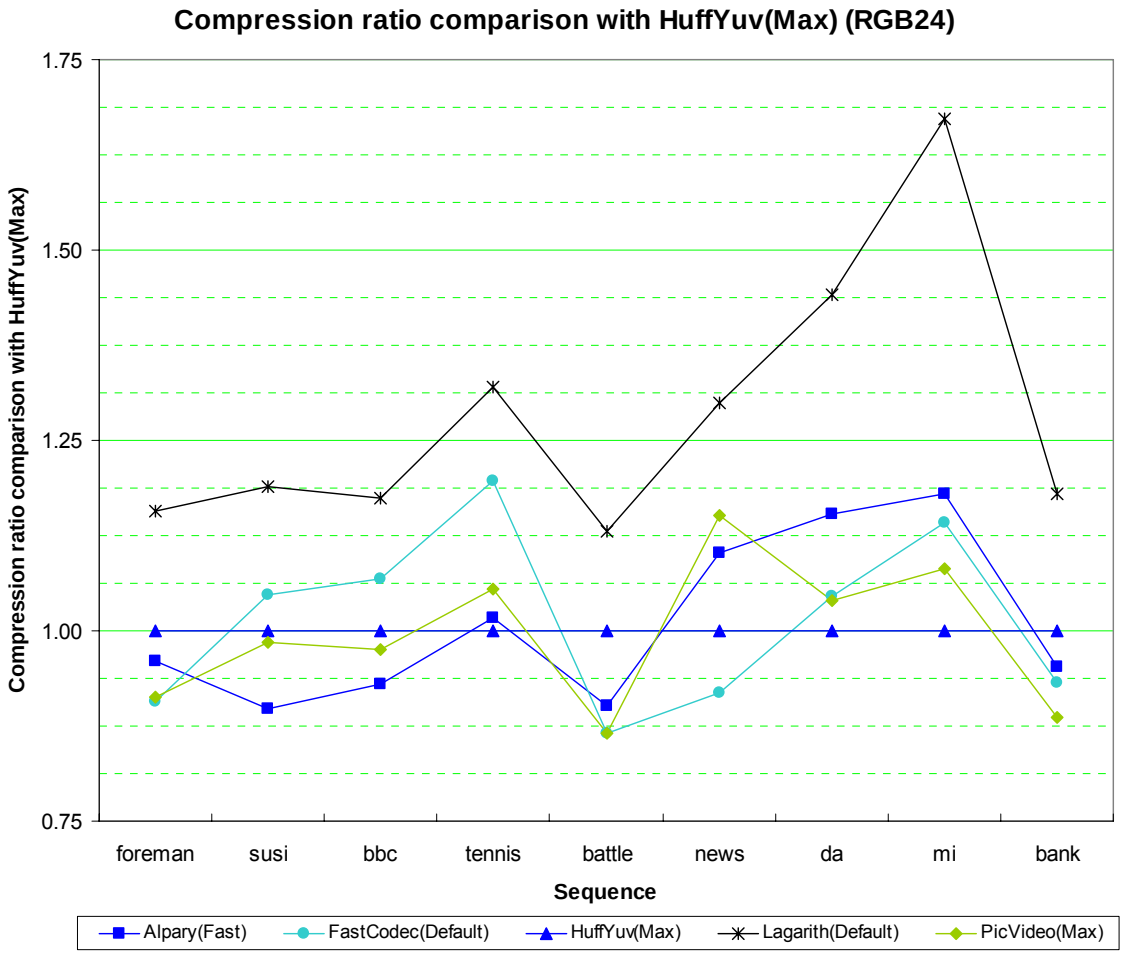


Figure 47. Video Capture: Relative per-file compession ratio for RGB24 in comparison to HuffYuv

Figure 48 and Figure 49 disclose the origin of Lagarith's leadership in speed. It processes files with large relatively monotonous areas more efficiently than its competitors. "Battle" video presents a problem for Lagarith in terms of encoding speed.

As for other codecs, their per-file encoding speed is generally similar with the exception of "Tennis" and noisy "News".

http://www.compression.ru/video/

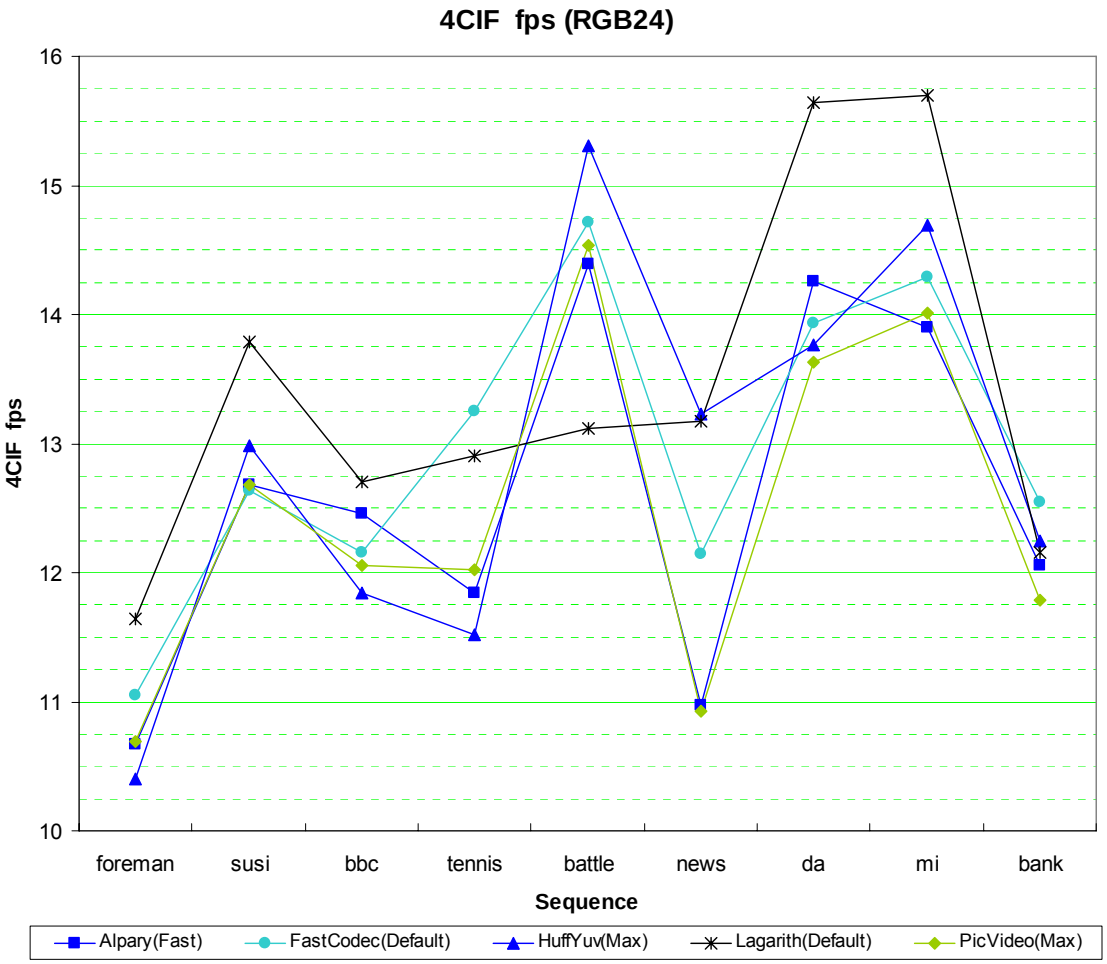


Figure 48. Video Capture: Per-file 4CIF fps for RGB24

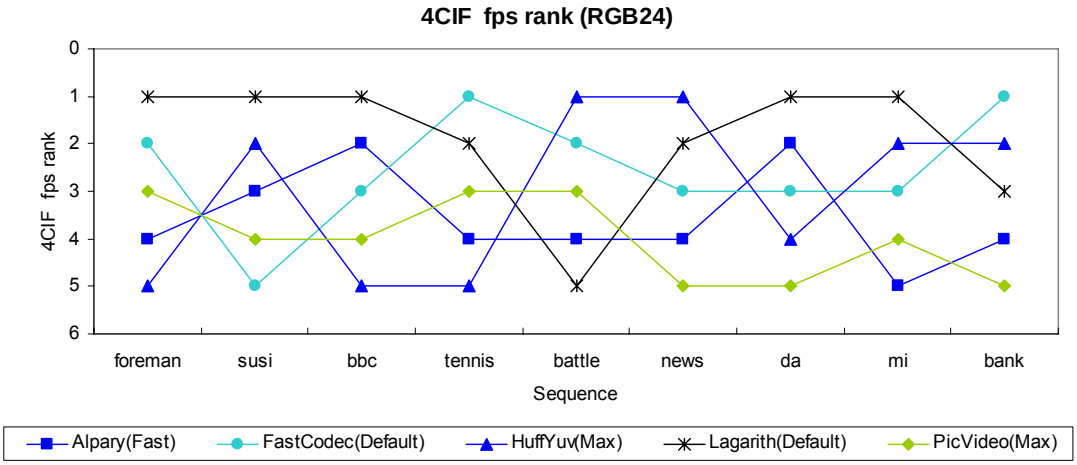


Figure 49. Video Capture: Per-file 4CIF fps ranking for RGB24

http://www.compression.ru/video/

4.3.2 YUY2

For this color space the situation is not as obvious as for RGB24, see Figure 50. **Lagarith** is the definite **leader by compression** efficiency, but it is somewhat slower than HuffYuv – 3% lower encoding speed. Compression ratio for Lagarith is 5% better than for HuffYuv. On the whole, if there is no a strict priority for encoding speed, the choice to use **Lagarith is preferable**.

ArithYuv, despite implementing theoretically better compression method, showed worse results for our test set than its direct competitor HuffYuv. Alpary and FastCodec are clearly underdogs in our comparison, being slower than other codecs and providing considerably worse compression ratio.

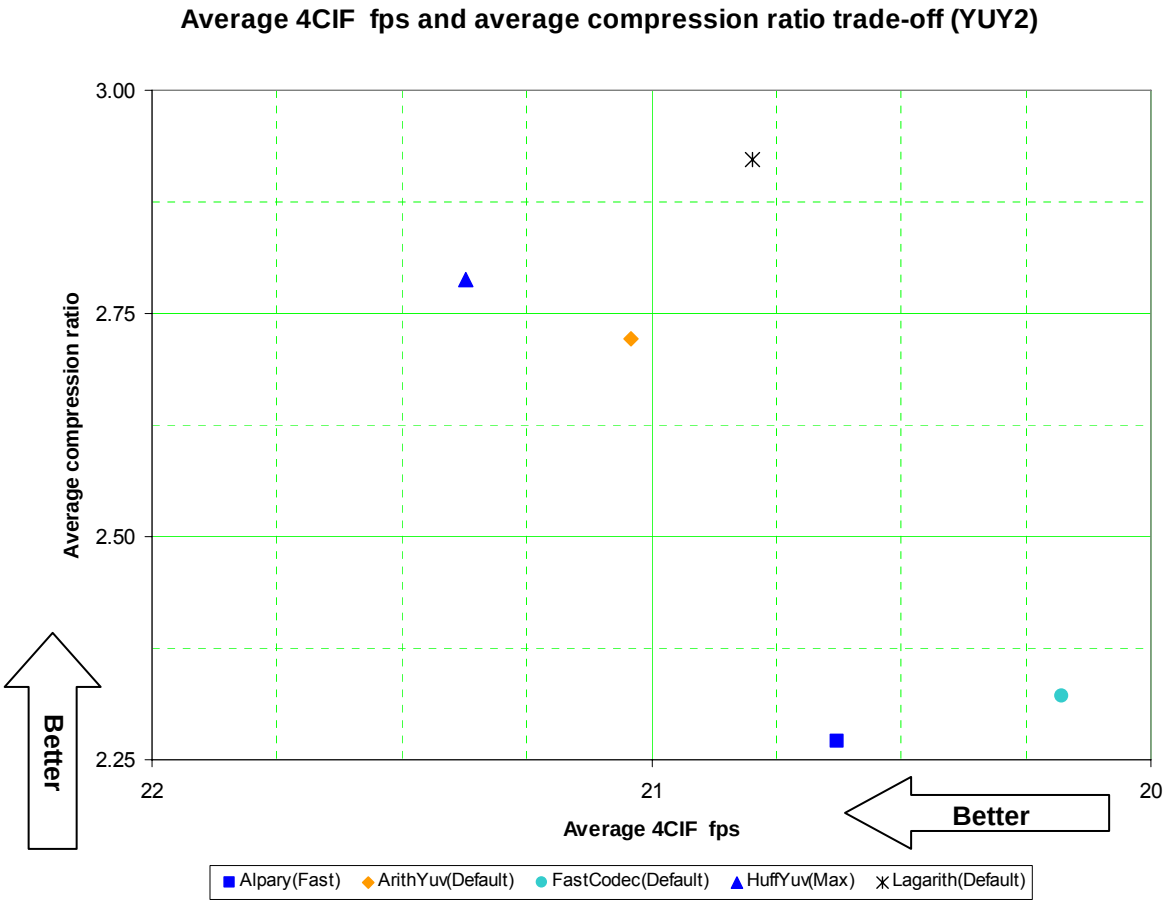


Figure 50. Video Capture: Average 4CIF fps and average compression ratio trade-off for YUY2

So the conclusions for YUY2 are:

- use Lagarith by default;
- use HuffYuv if encoding speed is really very important.

Figure 51 and Figure 52 show compression ratio only. As it can be seen from Figure 52, per-file results for compression are generally quite stable and predictable.

http://www.compression.ru/video/

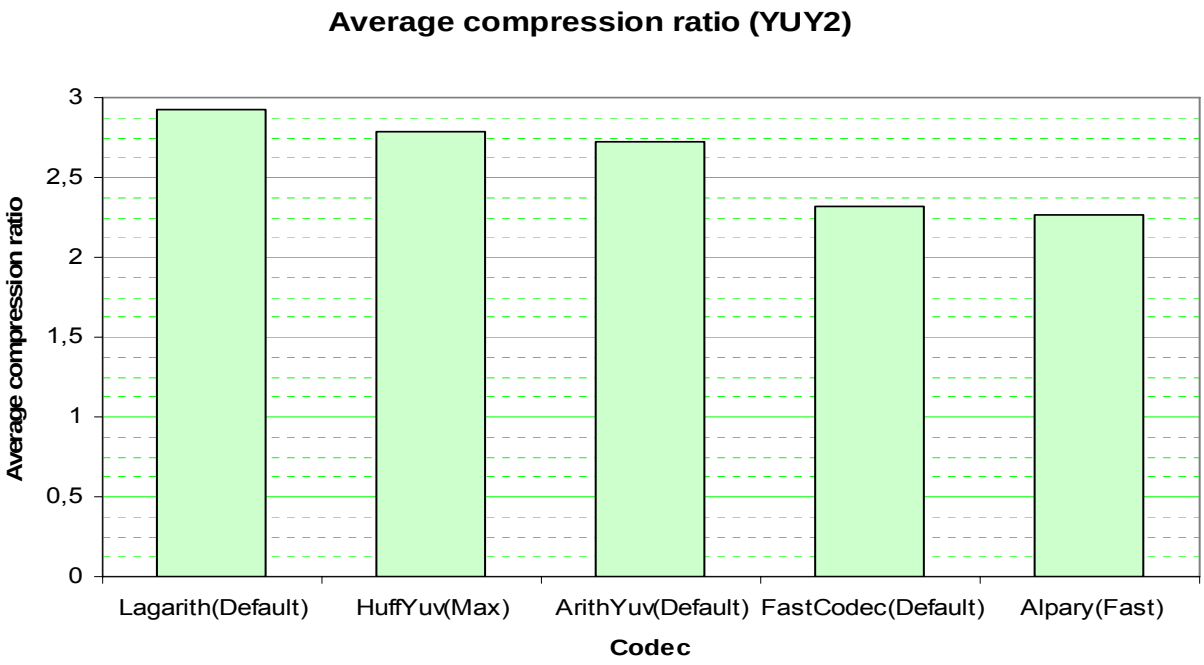


Figure 51. Video Capture: Average compression ratio for YUY2

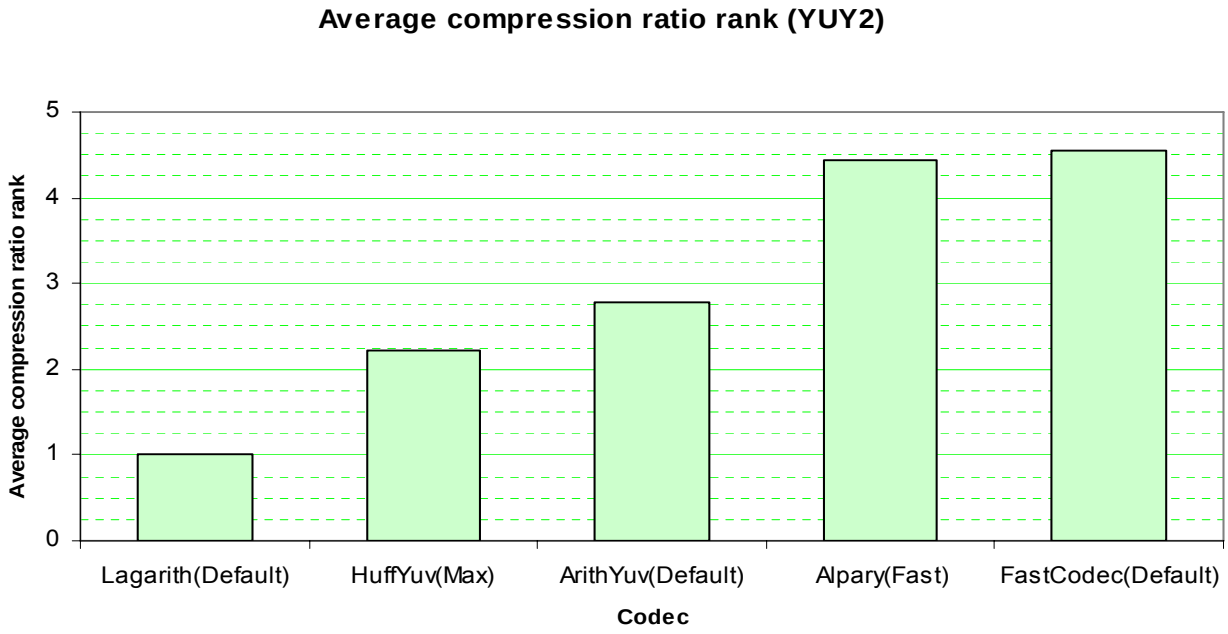


Figure 52. Video Capture: Average compression rank for YUY2

There is a similar situation for encoding speed (Figure 53, Figure 54).

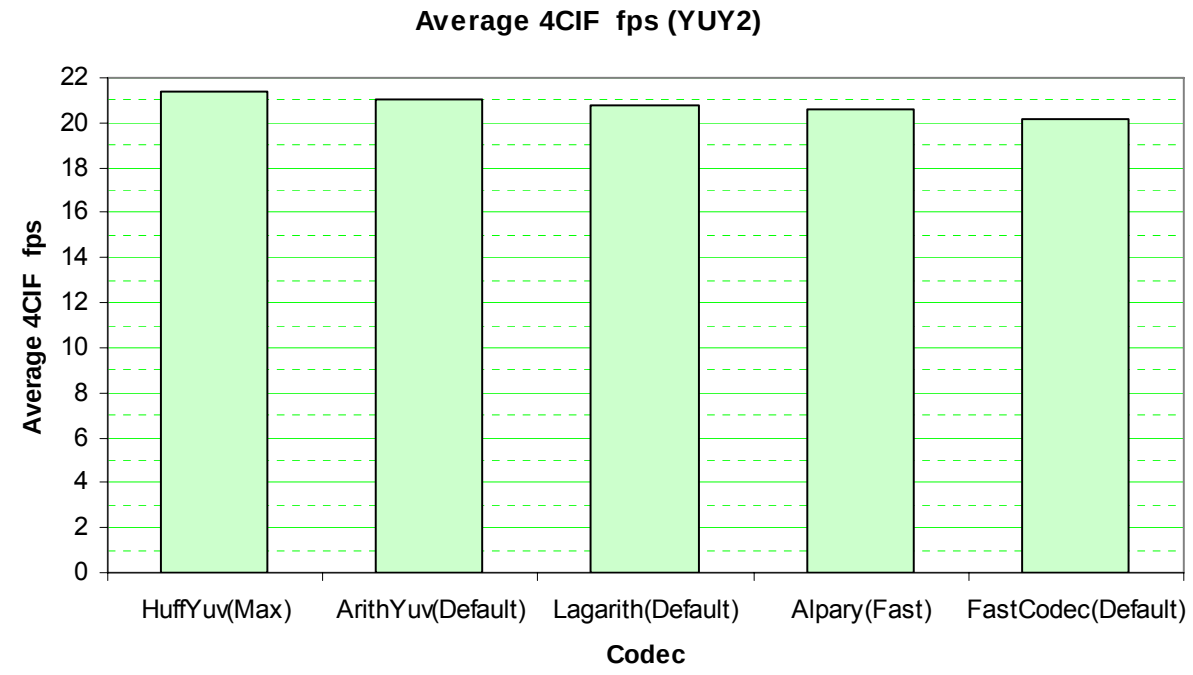


Figure 53. Video Capture: Average 4CIF fps for YUY2

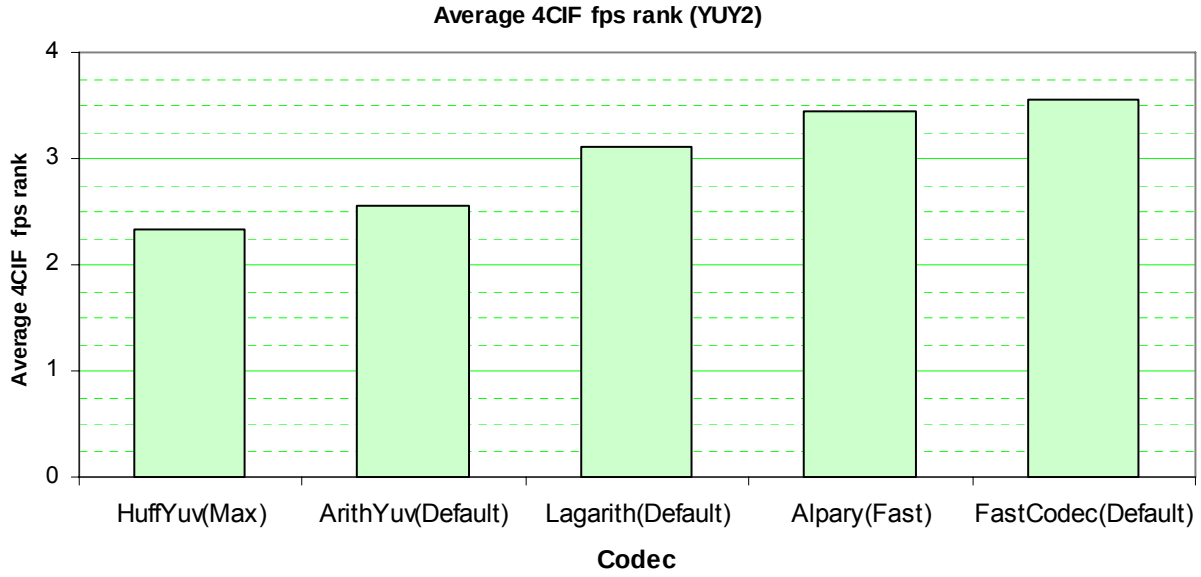


Figure 54. Video Capture: Average 4CIF fps rank for YUY2

As for RGB24, Lagarith significantly outperform other codecs by compression ratio on “DA” and “MI” videos having a lot of homogeneous regions without details (Figure 55- Figure 56, see also Figure 57). Fastcodec typically outperforms Alpary “Fast” on standard sequences like “BBC”, “News”, “Tennis”, but provides practically the same compression ratio for movie sequences.

http://www.compression.ru/video/

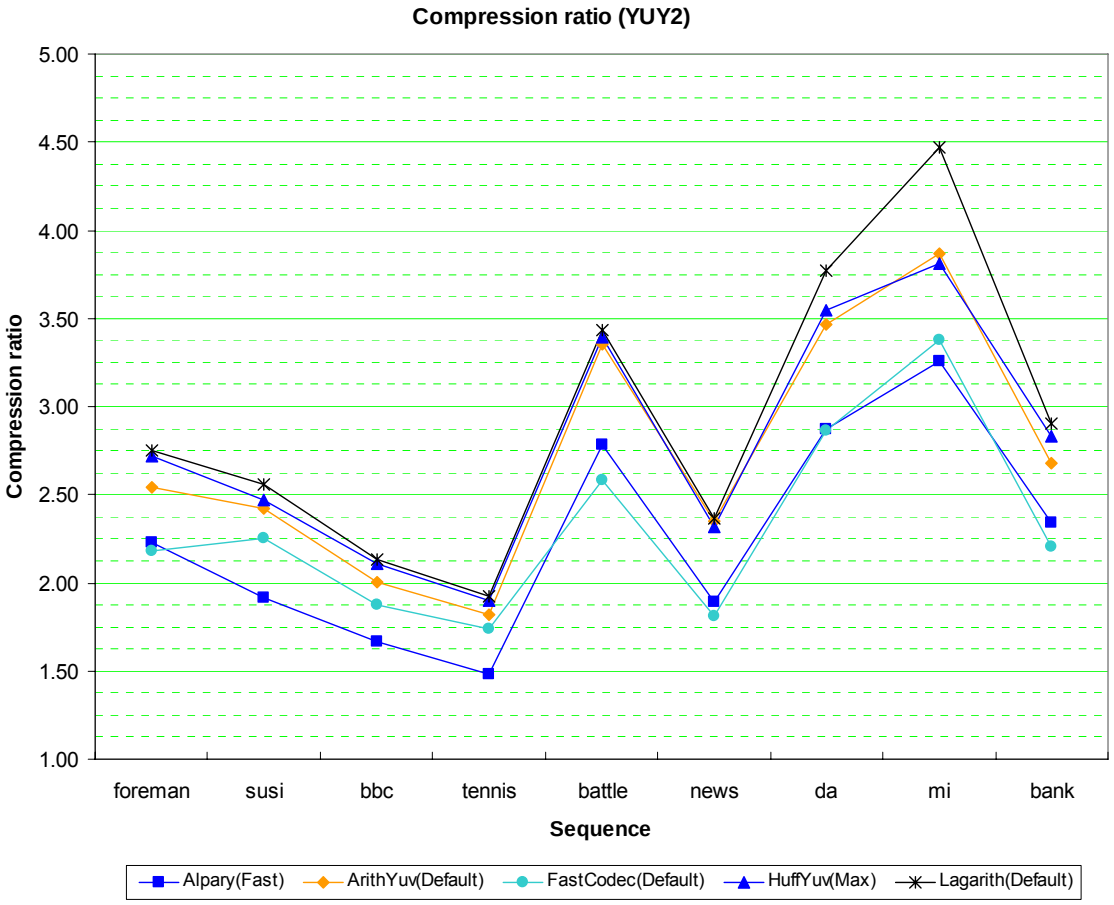


Figure 55. Video Capture: Compression ratio for each sequence in YUY2

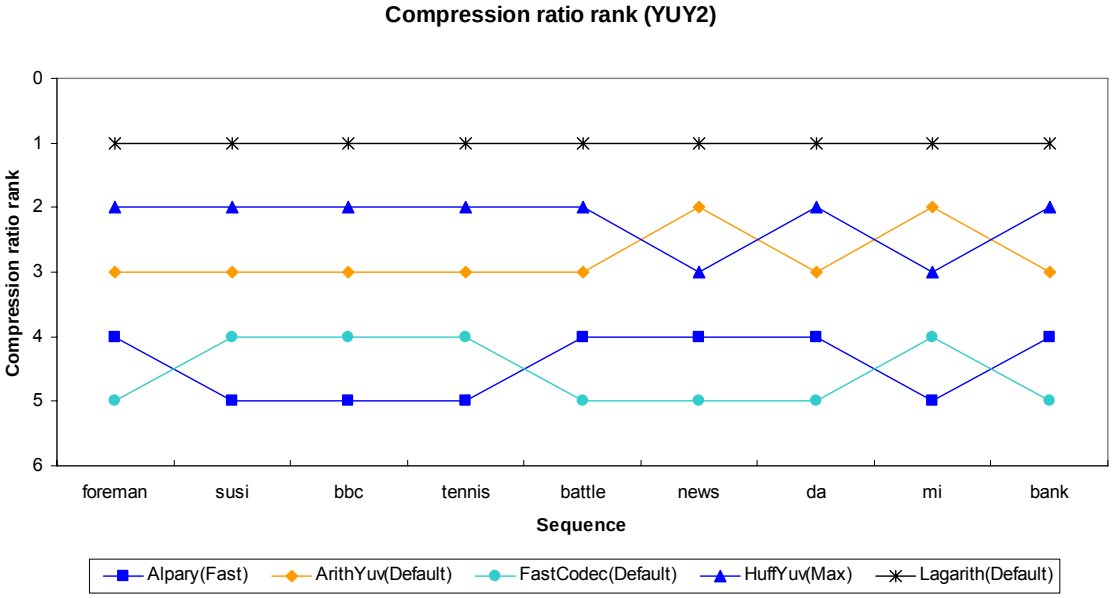


Figure 56. Video Capture: Per-file compression ratio ranking for YUY2

http://www.compression.ru/video/

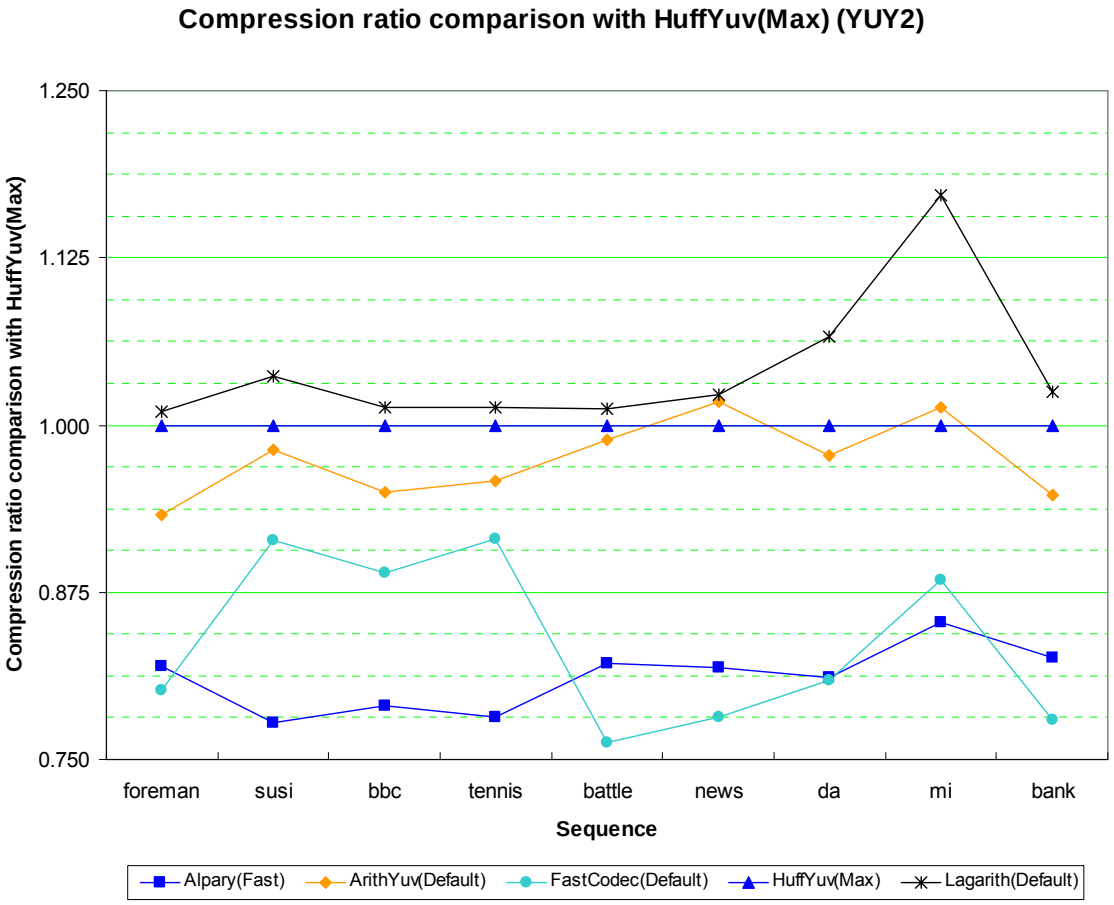


Figure 57. Video Capture: Relative per-file compession ratio for YUY2 in comparison to HuffYuv

Per-file encoding speed is also predictable in general, as it can be seen on Figure 58 and Figure 59. The only small surprises are a relative fall of encoding speed of HuffYuv on “MI” and an untypical change of speed on “Susi” and “BBC” for Alpary. It looks like that Lagarith has the smallest deviation of 4CIF fps for our test set.

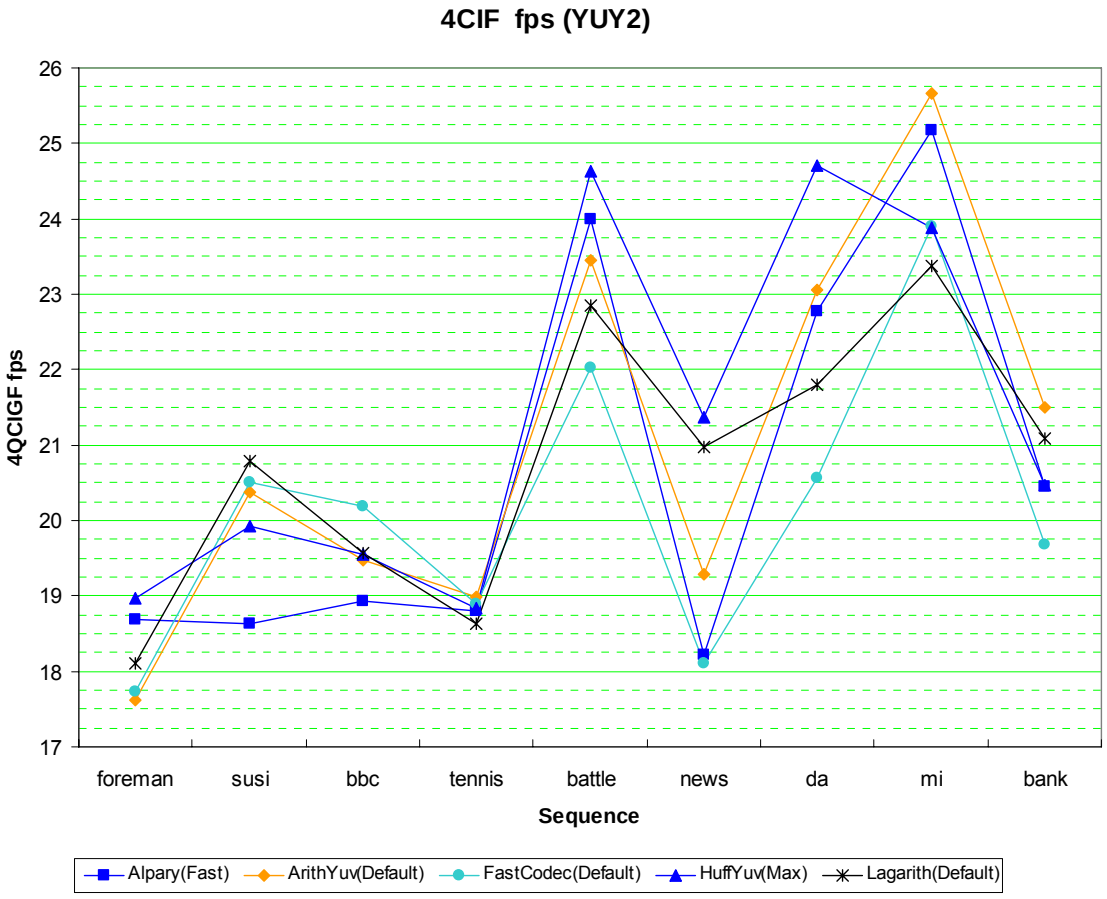


Figure 58. Video Capture: Per-file 4CIF fps for YUY2

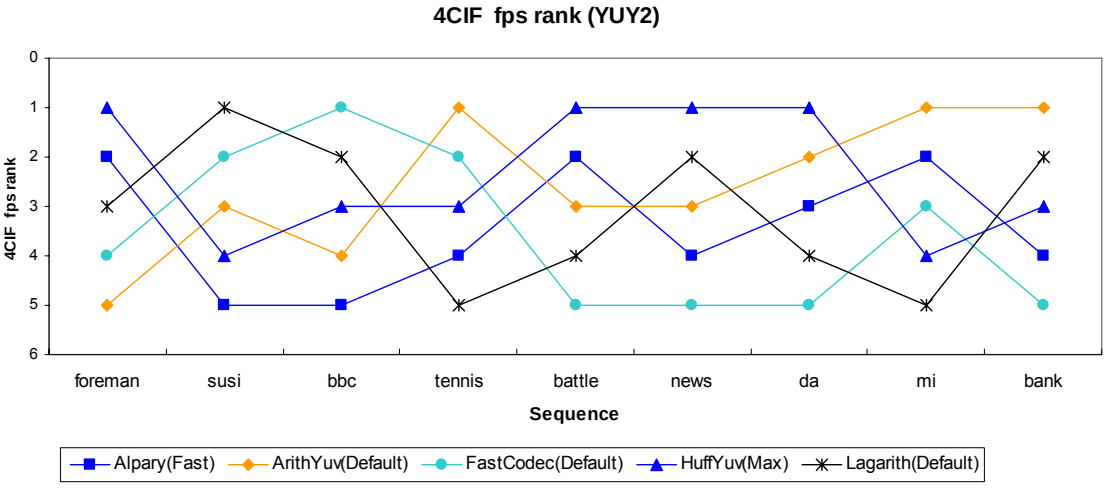


Figure 59. Video Capture: Per-file 4CIF fps ranking for YUY2

4.3.3 YV12

Results for YV12 are similar to those for YUY2. **Lagarith** is the **leader by compression** efficiency, but it is slightly slower than HuffYuv – 4% lower encoding speed. Compression ratio for Lagarith is 2.4% better than for HuffYuv. So difference by compression is not obvious, and the choice between HuffYuv and Lagarith depends on encoding time special requirements and type of content.

Alpary and LZO are clearly left behind by Lagarith and HuffYuv using both criteria chosen for this comparison.

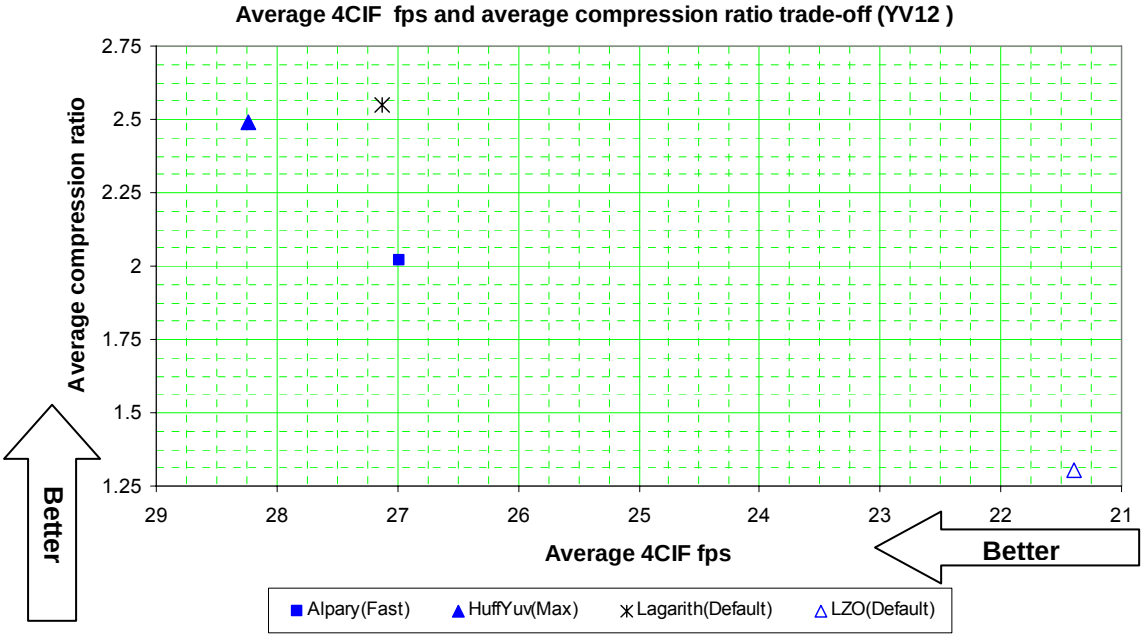


Figure 60. Video Capture: Average 4CIF fps and average compression ratio trade-off for YY12

The recommendations for YV12 are:

- use HuffYuv or Lagarith by default, at that use Lagarith if it provides significantly better compression for your files.

Results for compression ratio only and encoding speed only are presented on Figure 61 – Figure 64.

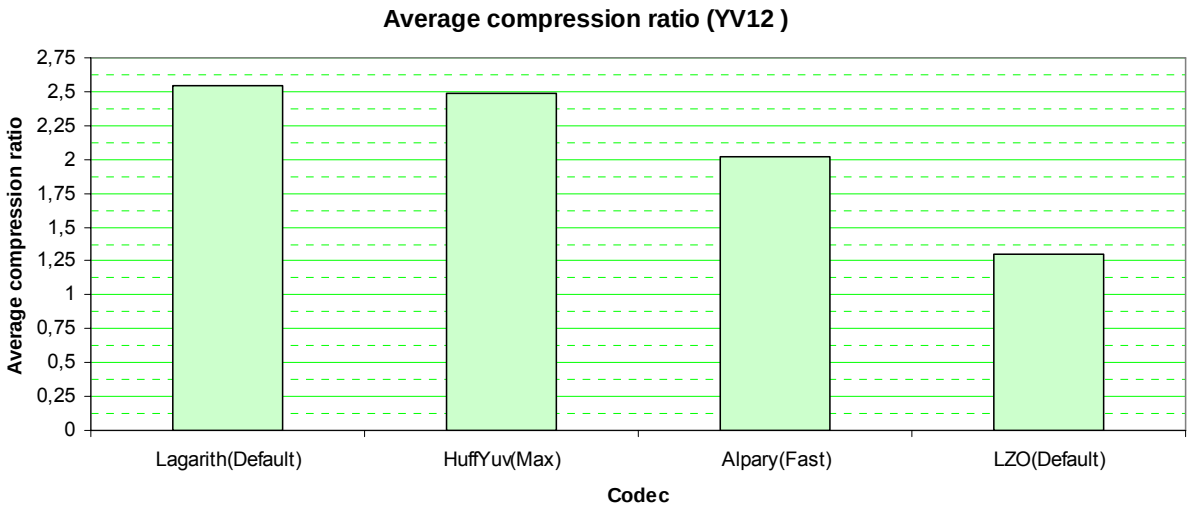


Figure 61. Video Capture: Average compression ratio for YV12

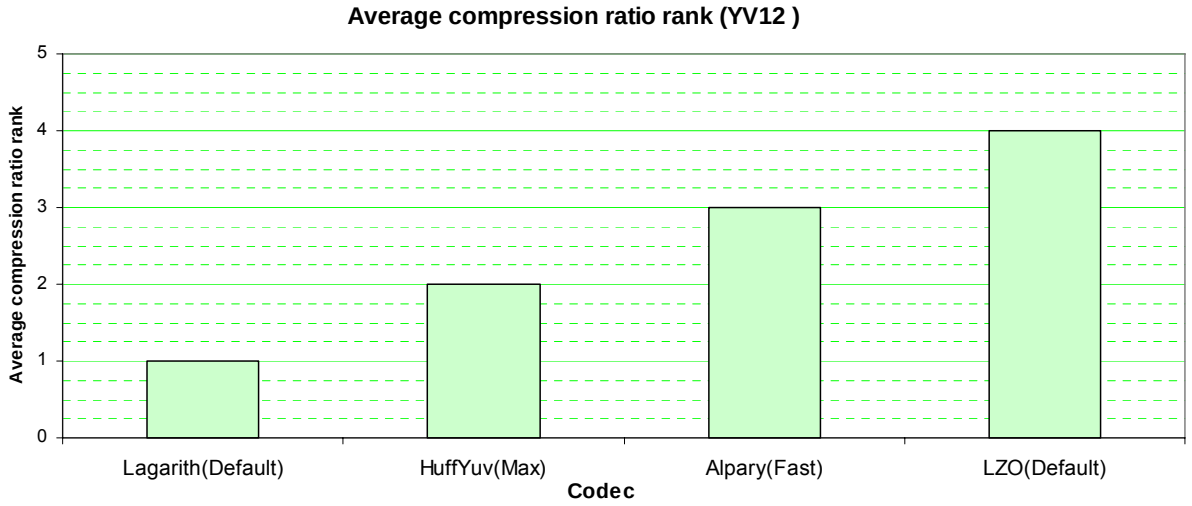


Figure 62. Video Capture: Average compression rank for YV12

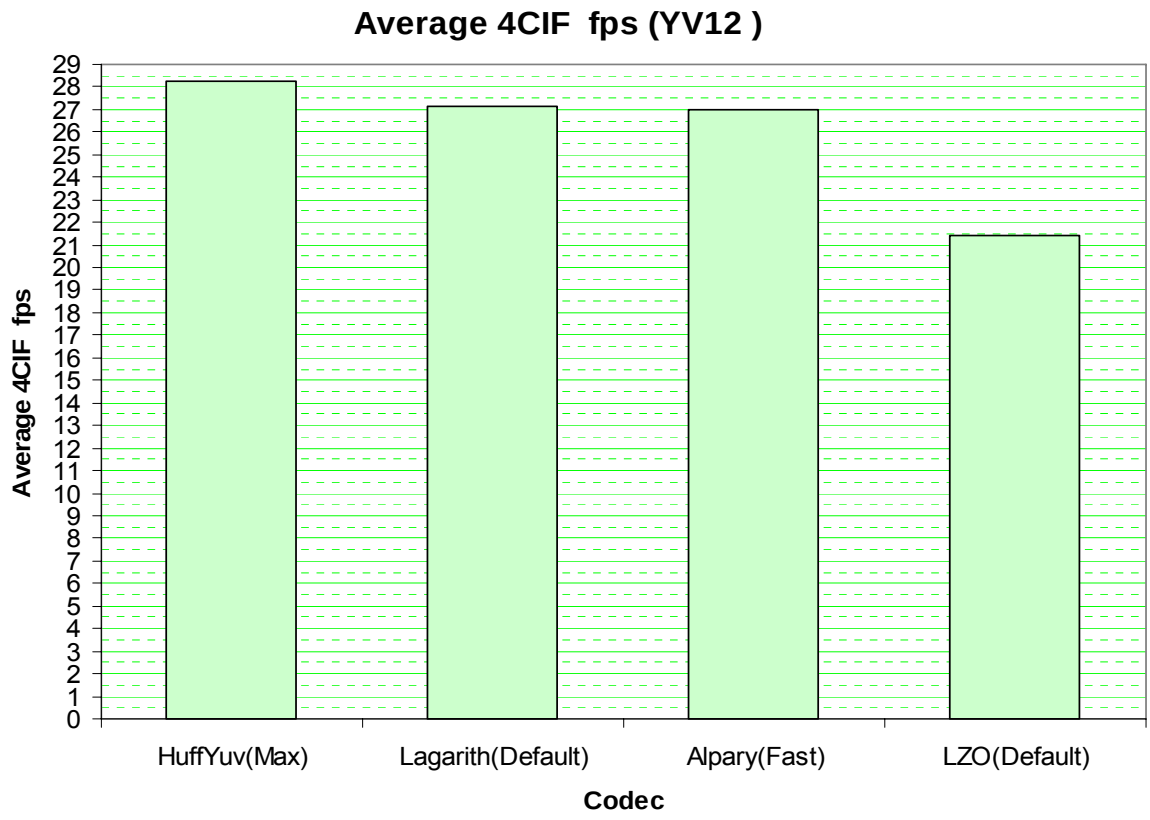


Figure 63. Video Capture: Average 4CIF fps for YV12

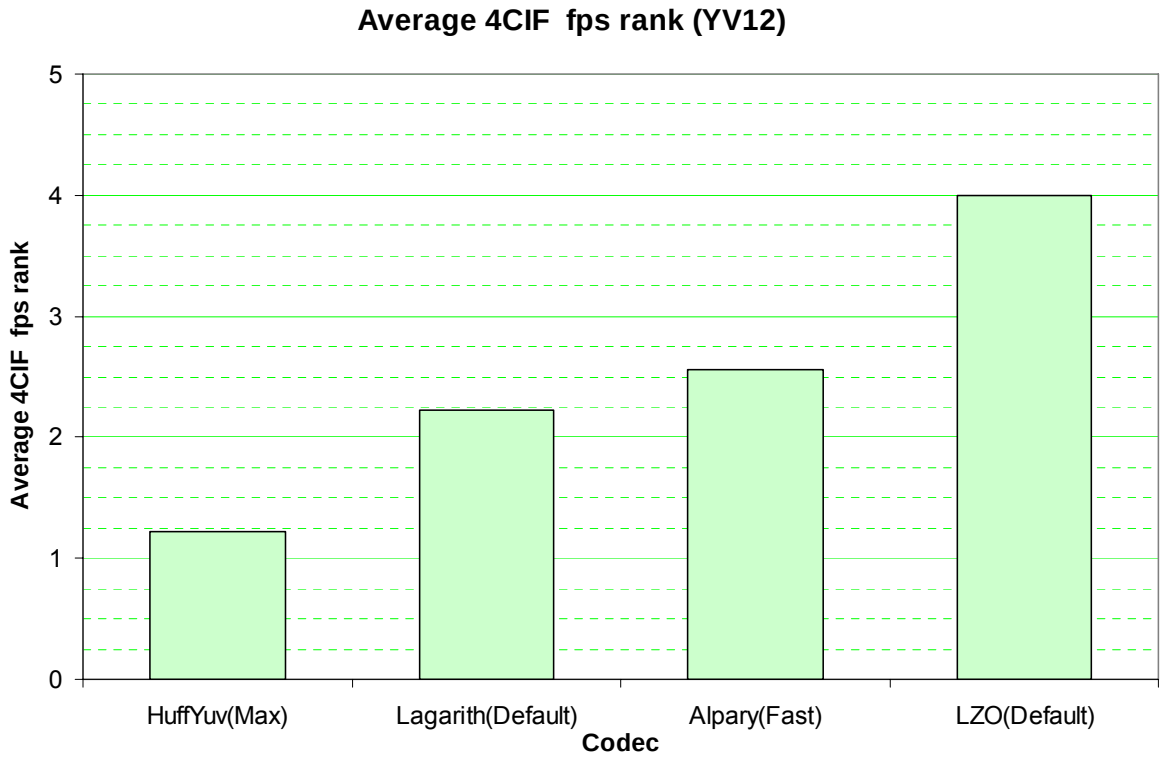


Figure 64. Video Capture: Average 4CIF fps rank for YV12

http://www.compression.ru/video/

Results in compression for individual sequences correlate very high with average resuls. As Figure 65 - Figure 66 show, results are very predictable. Compression rank of each codec is constant for the entire test set.

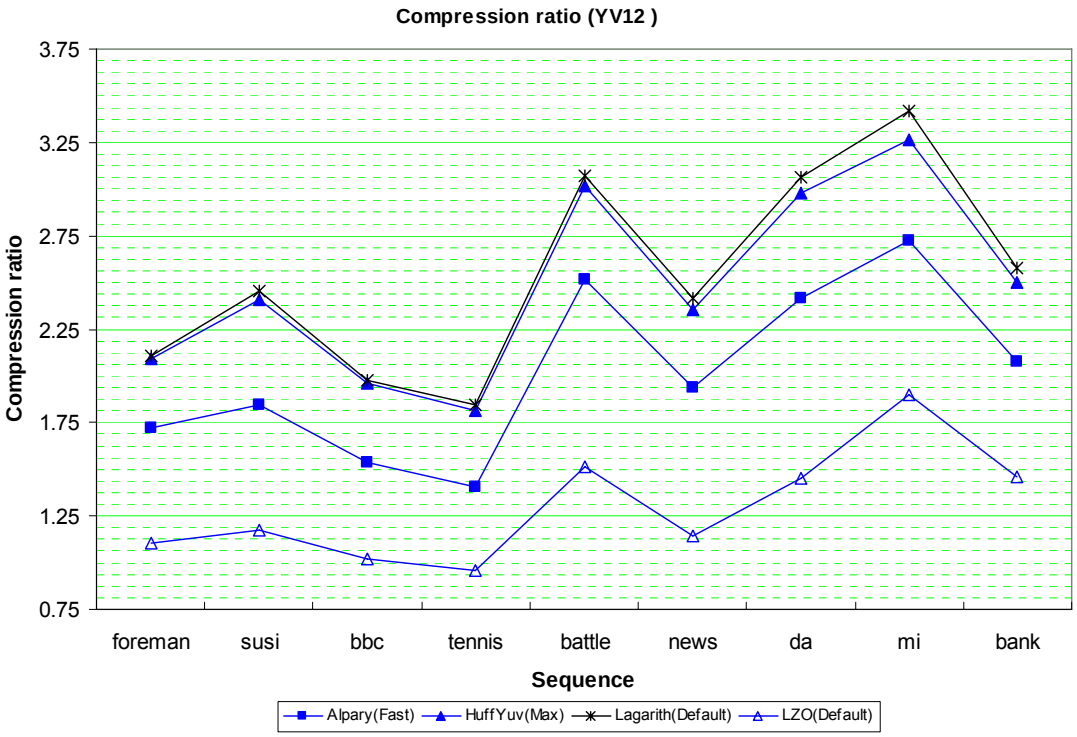


Figure 65. Video Capture: Compression ratio for each sequence in YV12

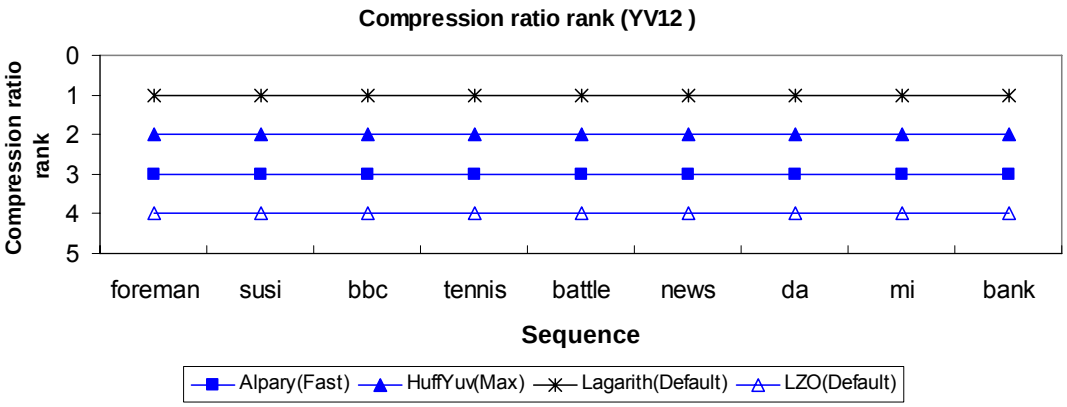


Figure 66. Video Capture: Per-file compression ratio ranking for YV12

It should be noted that in contrast to RGB24 and YUY2, there are no files where compression ratio for Lagarith is considerably better than for HuffYuv. It means that improvements implemented in Lagarith are likely give a boost for more smooth color planes only in general, not for luma planes.

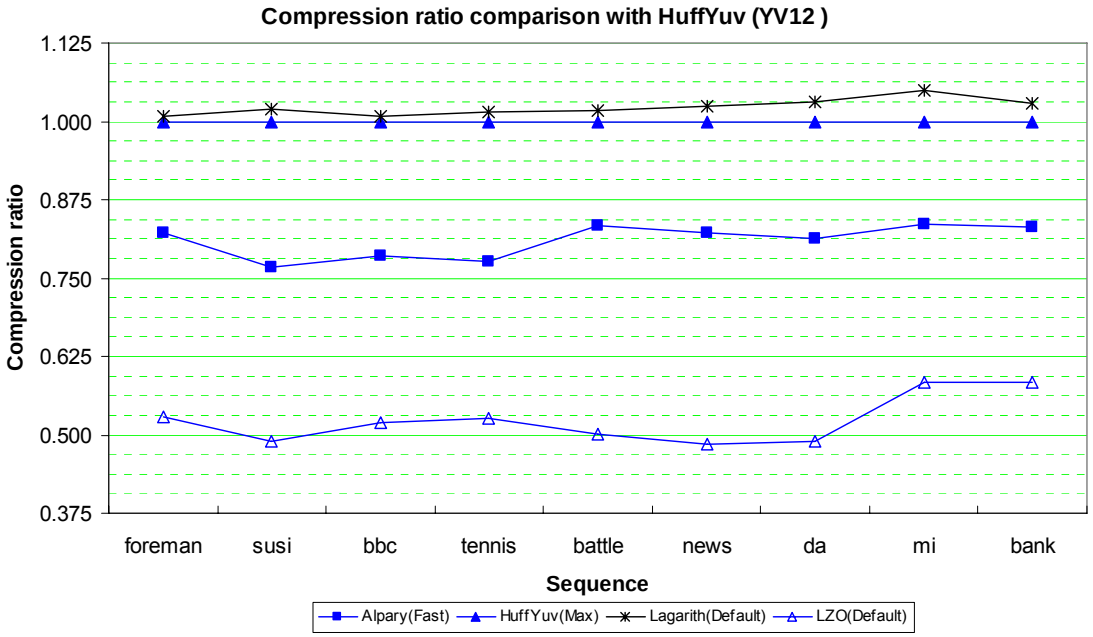


Figure 67. Video Capture: Relative per-file compession ratio for RGB24 in comparison to YV12

Per-file encoding speed results do not contain any surprises also; see Figure 68 - Figure 69.

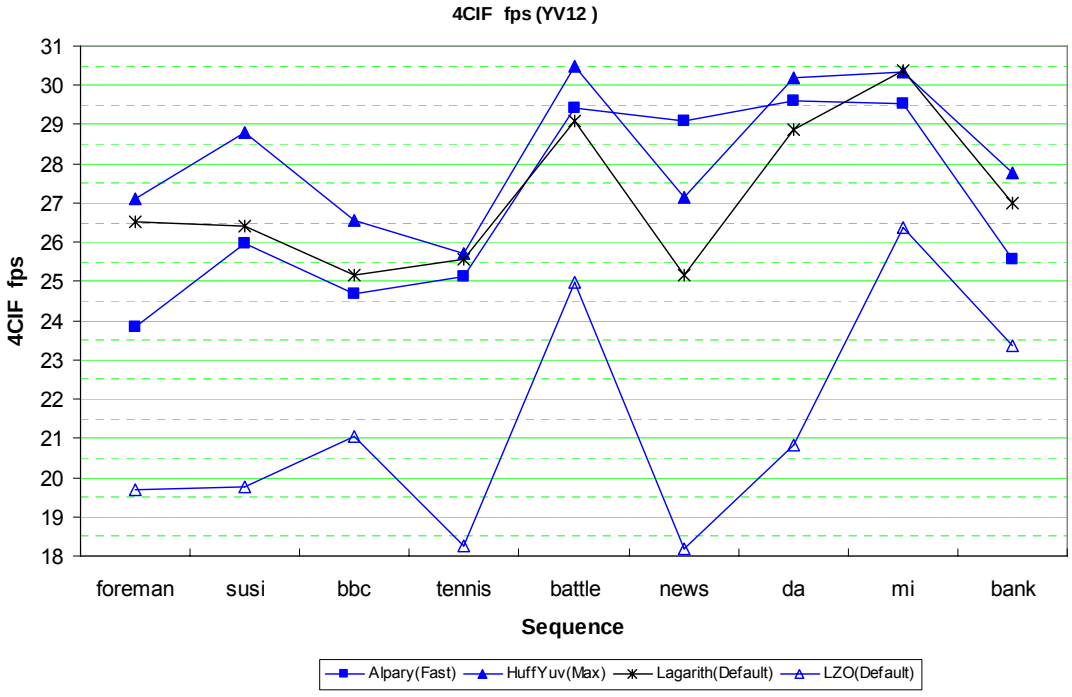


Figure 68. Video Capture: Per-file 4CIF fps for YV12

http://www.compression.ru/video/

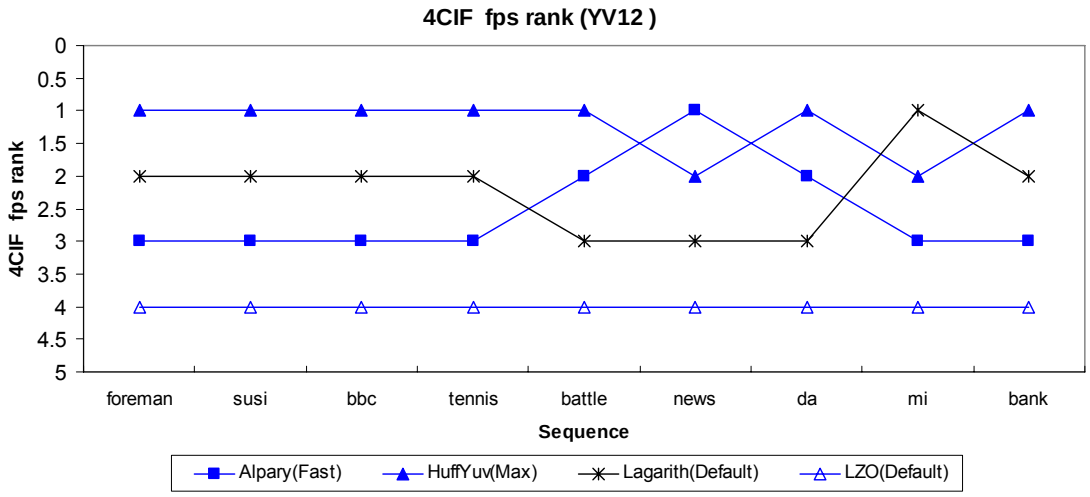


Figure 69. Video Capture: Per-file 4CIF fps ranking for YV12

4.3.4 Overall Rating

For some applications it may be useful to understand average ranking of codecs for all color spaces.

Only three codecs from this section work in all three color spaces we used. They are: Lagarith, HuffYuv and Alpary. They are compared by compression ratio and encoding speed ranks on Figure 70 - Figure 71.

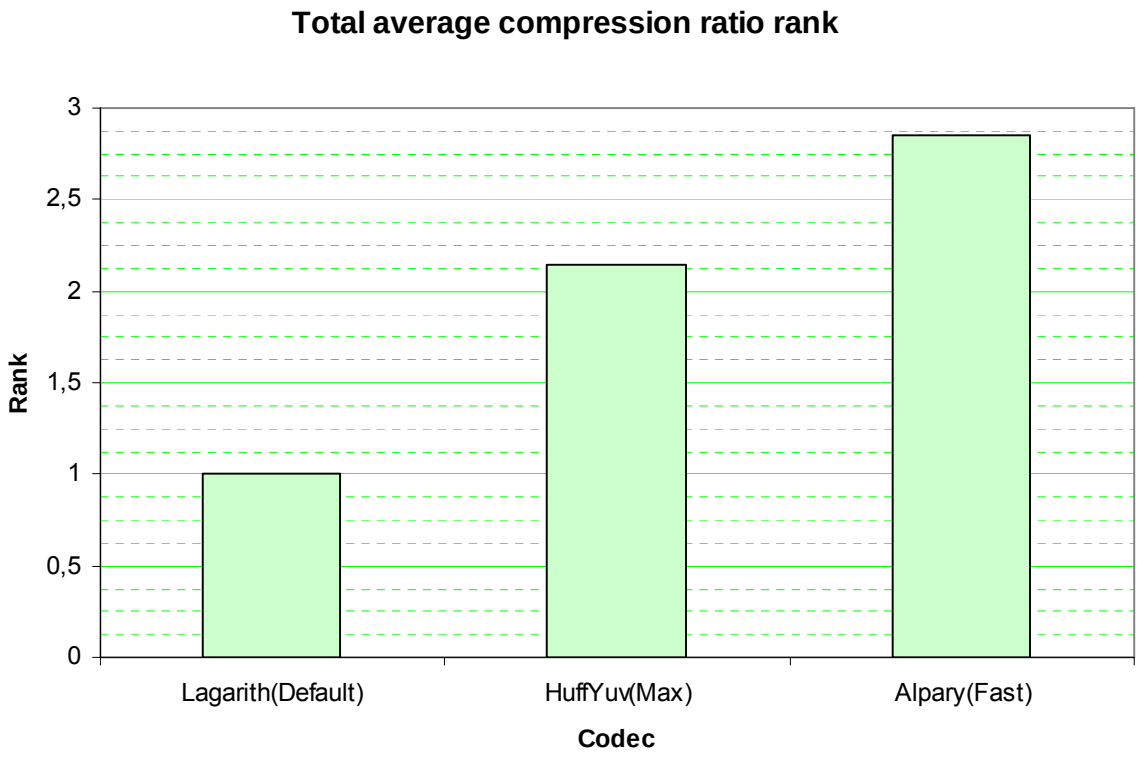


Figure 70. Video Capture: Total average compression ratio rank for RGB24, YUY2, YV12

http://www.compression.ru/video/

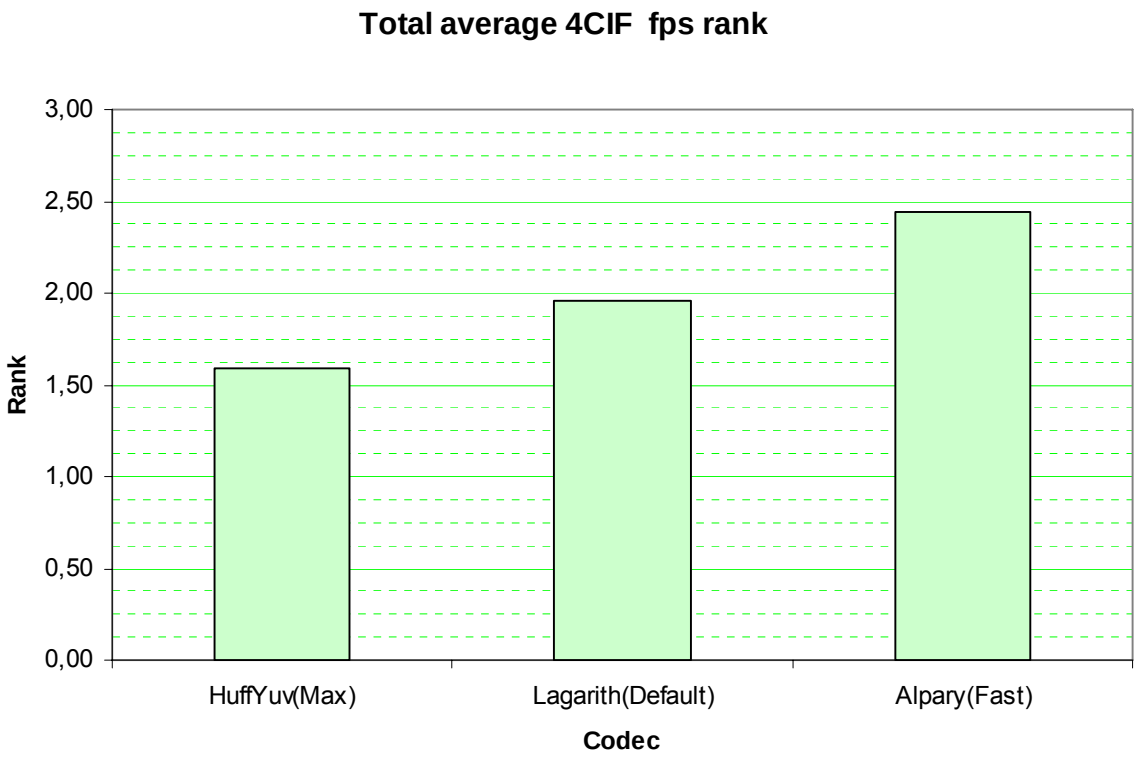


Figure 71. Video Capture: Total average 4CIF fps rank for RGB24, YUY2, YV12

So Lagarith is generally a natural choice as a balanced codec with better compression. If encoding speed is very important and video in YUY2 and/or in YV12 color space are to be processed, than it makes sense to consider HuffYuv.

4.4 Maximum Compression Area

4.4.1 RGB24

In RGB24 maximum compression for our test set is provided by YULS and FFV1, YULS being slightly better (0.2% on average), but almost 10 times slower for “Max” mode (Figure 72). The chart on Figure 74 shows that generally YULS gives better compression ratio per each file of our test set. However, it can be concluded that **practical leader** for this color space is **FFV1**, at that **YULS** being a choice to obtain **maximum compression** in general.

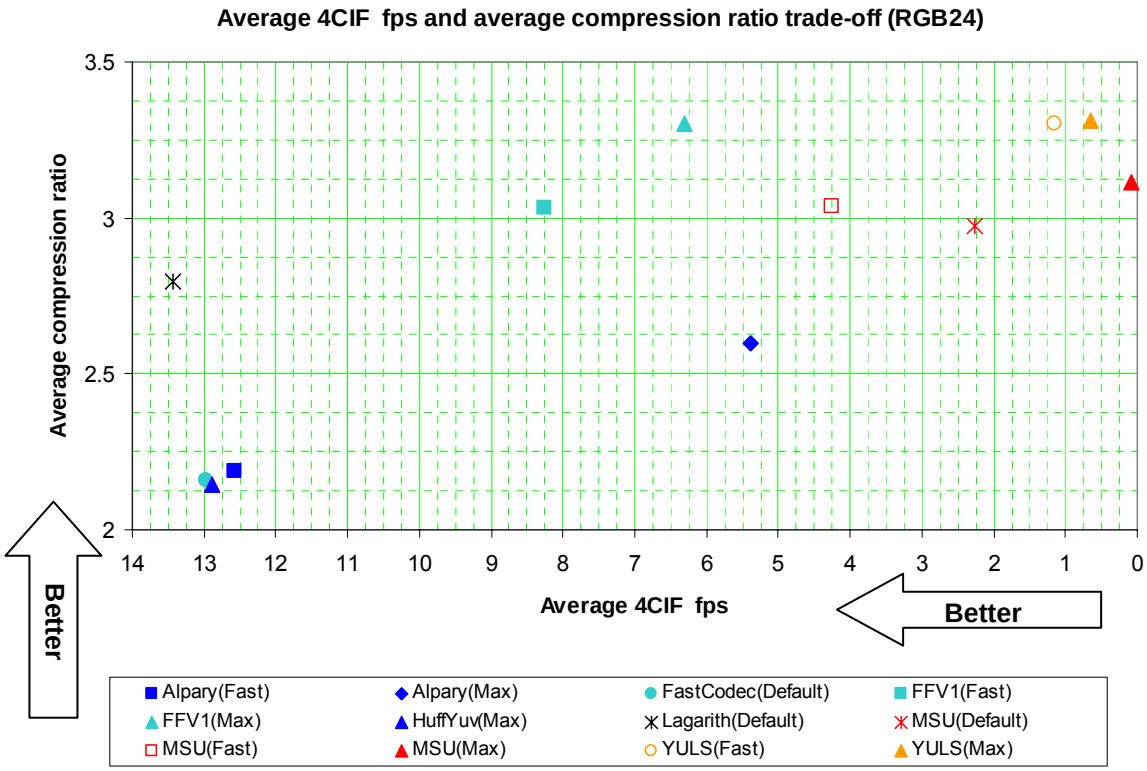


Figure 72. Maximum Compression: Average 4CIF fps and average compression ratio trade-off for RGB24

As it can be seen from Figure 72 - Figure 73, Alpary, FastCodec and MSU codec are covered by FFV1 and/or Lagarith by both criteria, so, if there are no other considerations, there are no reasons to use them in general for this color space.

In short, the following conclusions and suggestions can de drawn from RGB24 results:

- consider using YULS if compression ratio is everything;
- use FFV1 by default;
- use Lagarith if speed is also important.

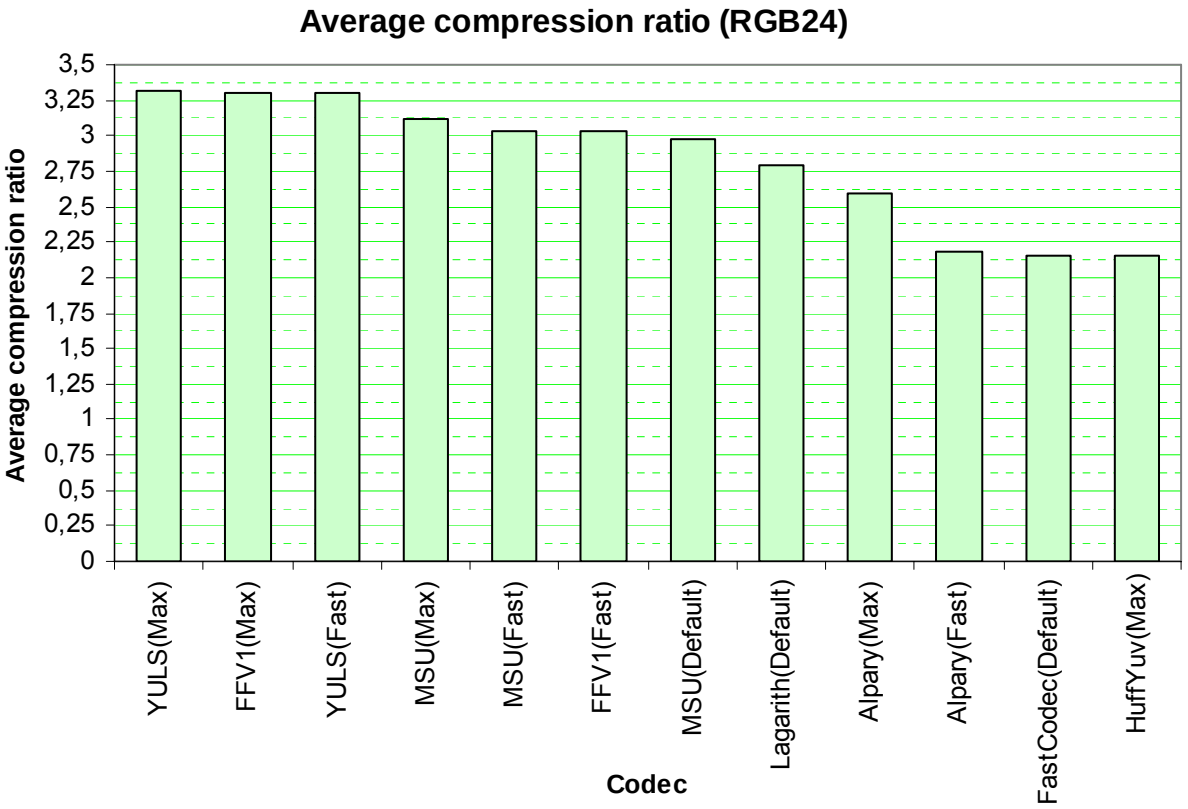


Figure 73. Maximum Compression: Average compression ratio for RGB24

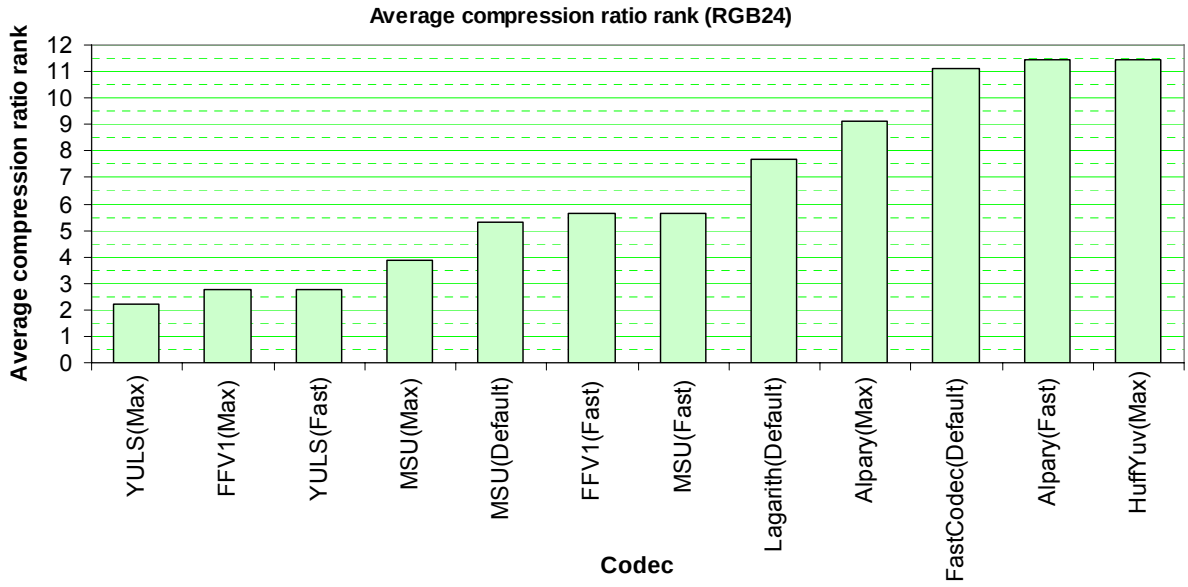


Figure 74. Maximum Compression: Average compression rank for RGB24

Figure 75 - Figure 76 highlight encoding speed results. There are 3 groups: fast Lagarith group, medium FFV1 group, and YULS constitutes a slow group. Also there is a MSU “Max” preset being totally impractical with less than 0.1 fps for 4CIF (about 10 times slower than even YULS)

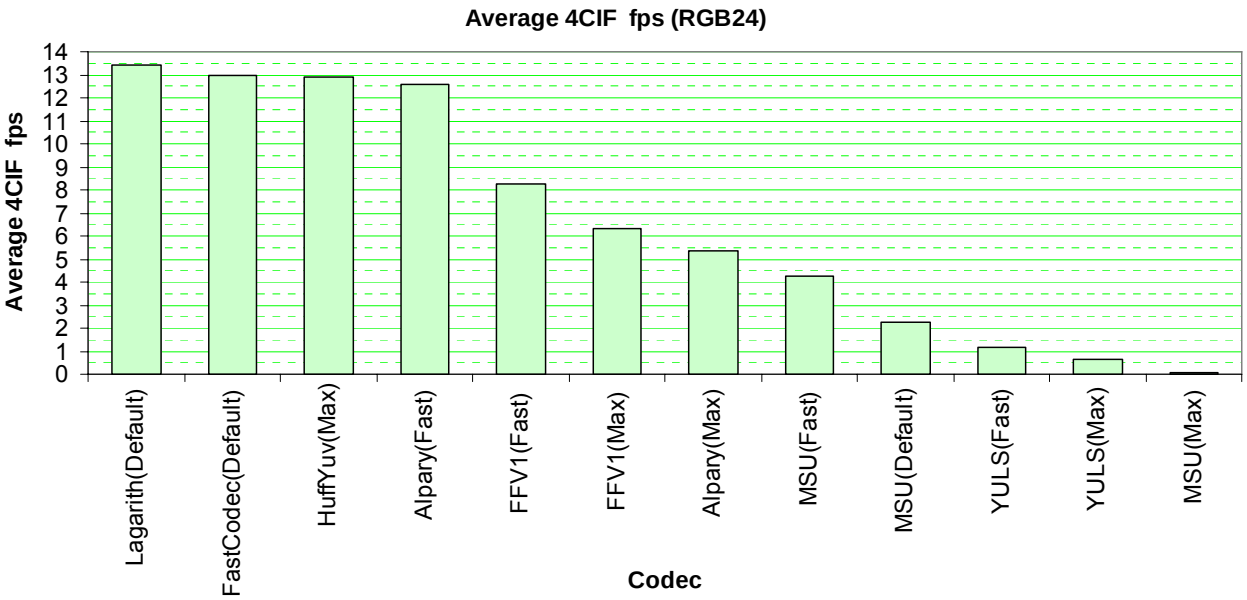


Figure 75. Maximum Compression: Average 4CIF fps for RGB24

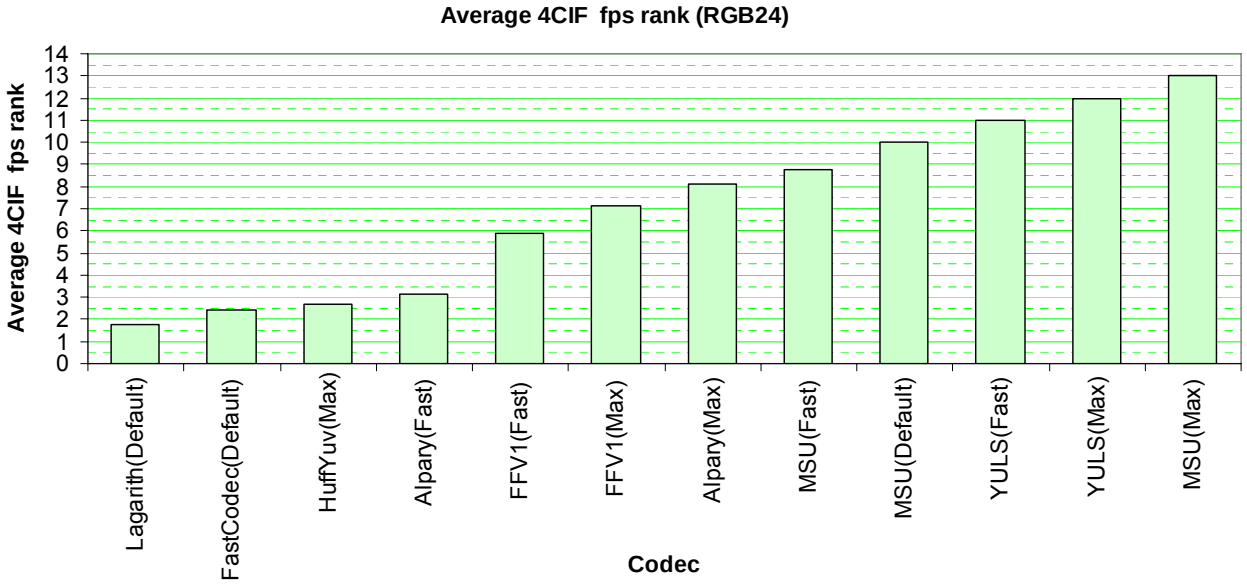


Figure 76. Maximum Compression: Average 4CIF fps rank for RGB24

Per-file results are presented on Figure 77 and Figure 78. As it can be seen, FFV1 behaves very well on “Battle” and “News”. On the contrary, YULS have problems with these two, especially “News”, providing stable results for other files. If one excludes “News”, a lead of YULS against FFV1 becomes 2% instead of 0.2%. It means that one should consider FFV1 and YULS results on files of interest to make a conclusion what to use in some specific case.

Also it can be noted that YULS and FFV1 compresses highly compressable “DA” and “MI” sequences far better than competitors. Another interesting fact is MSU superiority for “Tennis”. This is the only file it gives better results than FFV1 and YULS.

http://www.compression.ru/video/

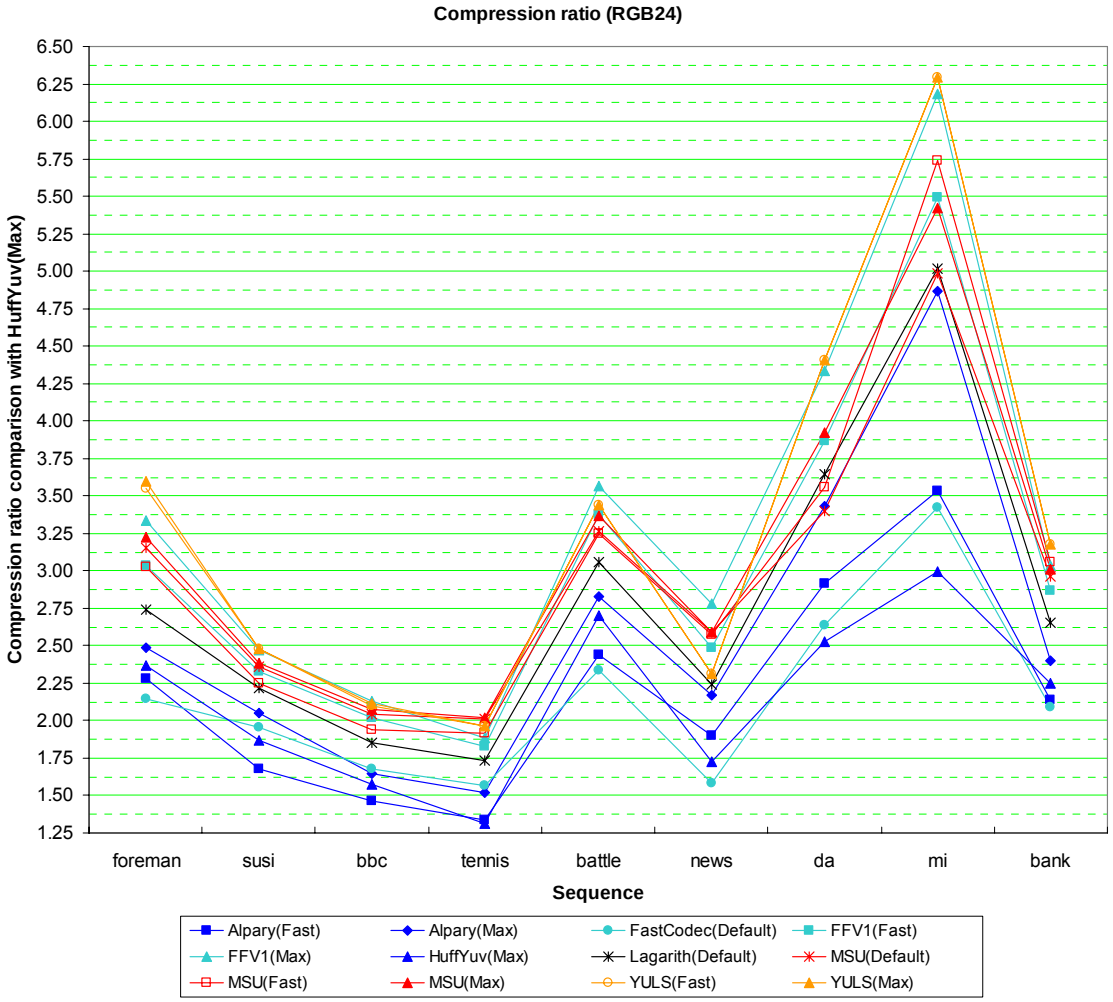


Figure 77. Maximum Compression: Compression ratio for each sequence in RGB24

http://www.compression.ru/video/

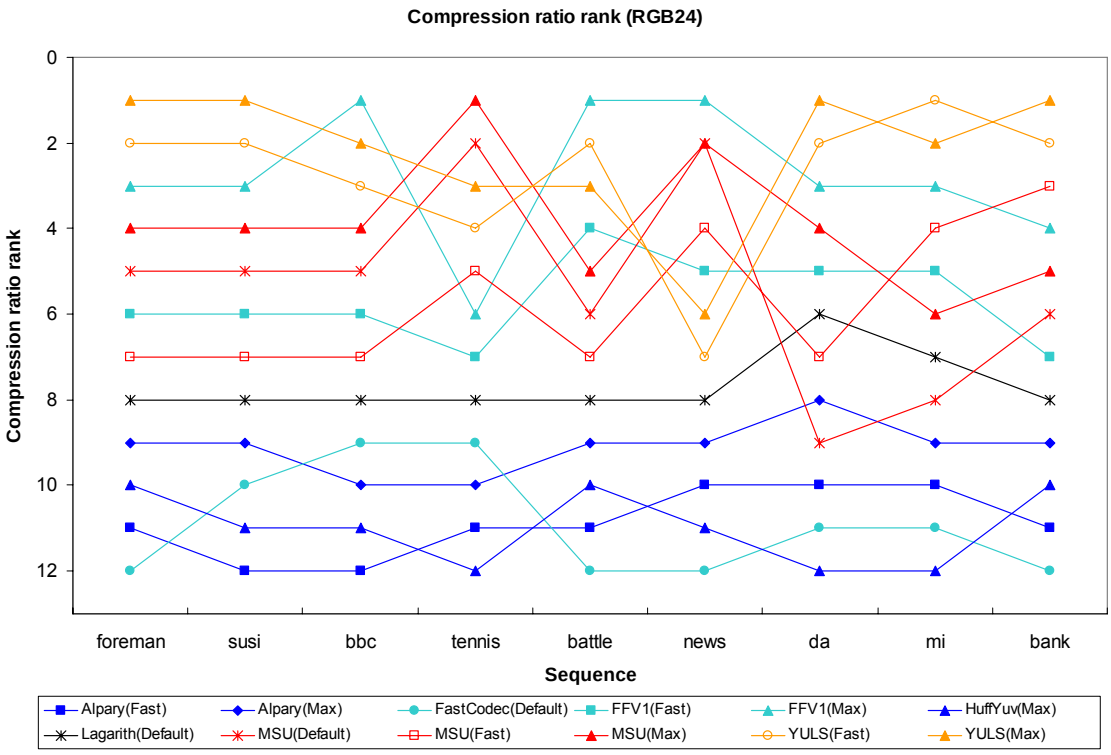


Figure 78. Maximum Compression: Per-file compression ratio ranking for RGB24

As it can be seen from Figure 79, advantage of complex codecs over HuffYuv may be as high as more than 2 times on certain files. It is 54% on average in RGB24 for our test set.

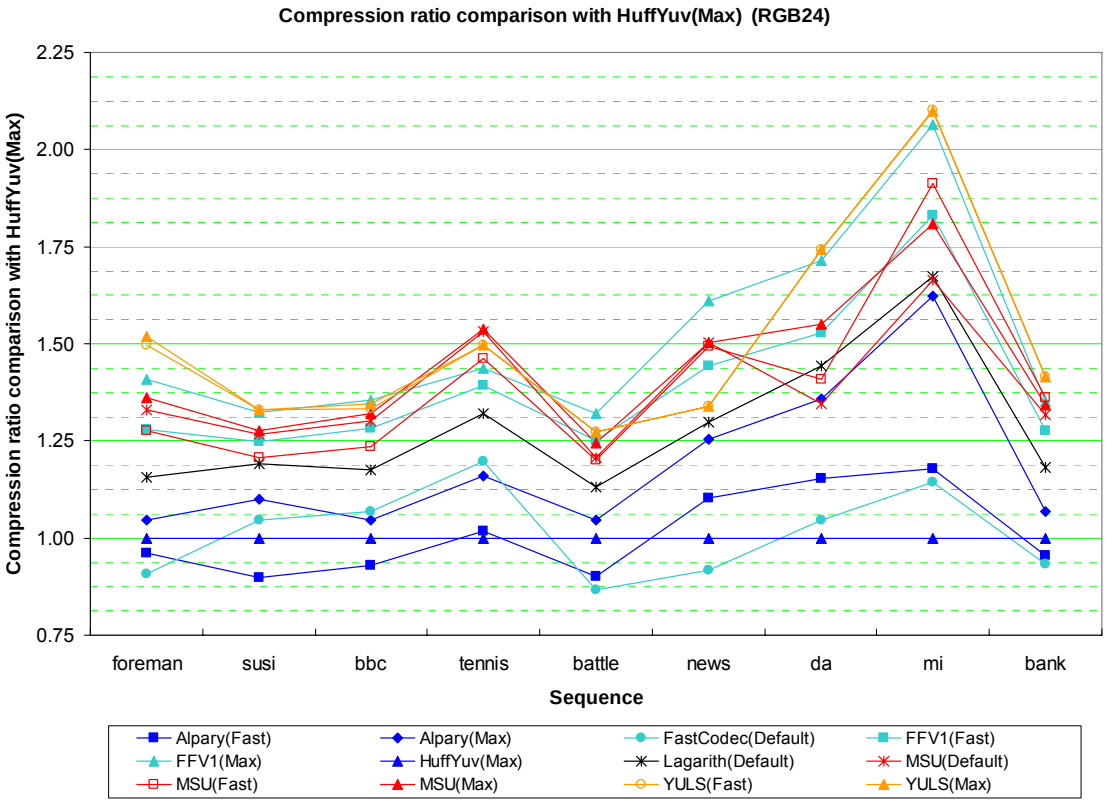


Figure 79. Maximum Compression: Relative per-file compession ratio for RGB24 in comparison to HuffYuv

Per-file encoding speed data visualized on Figure 80 - Figure 81 proof that there are may be significant encoding speed deviations, but generally results are predictable. Among relatively fast, but high compression codecs and presets stable results are being shown by FFV1 “Fast” and MSU “Default”.

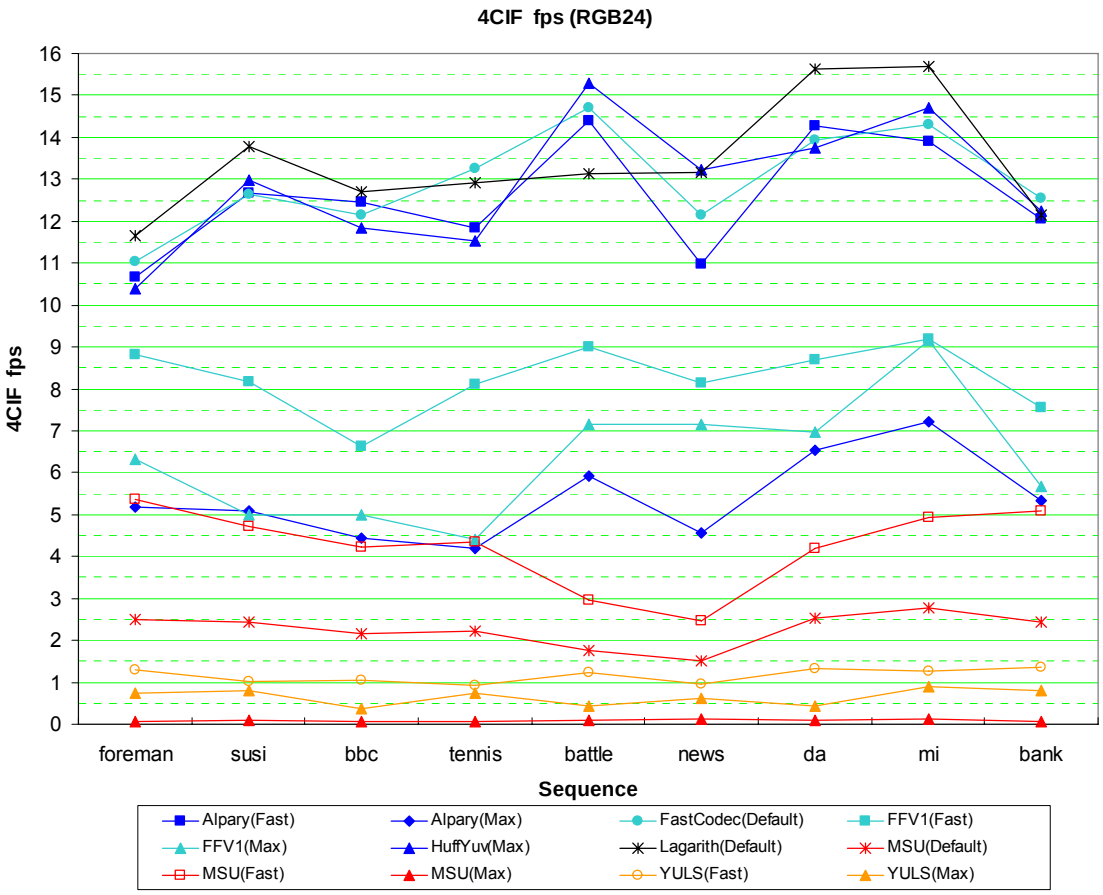


Figure 80. Maximum Compression: Per-file 4CIF fps for RGB24

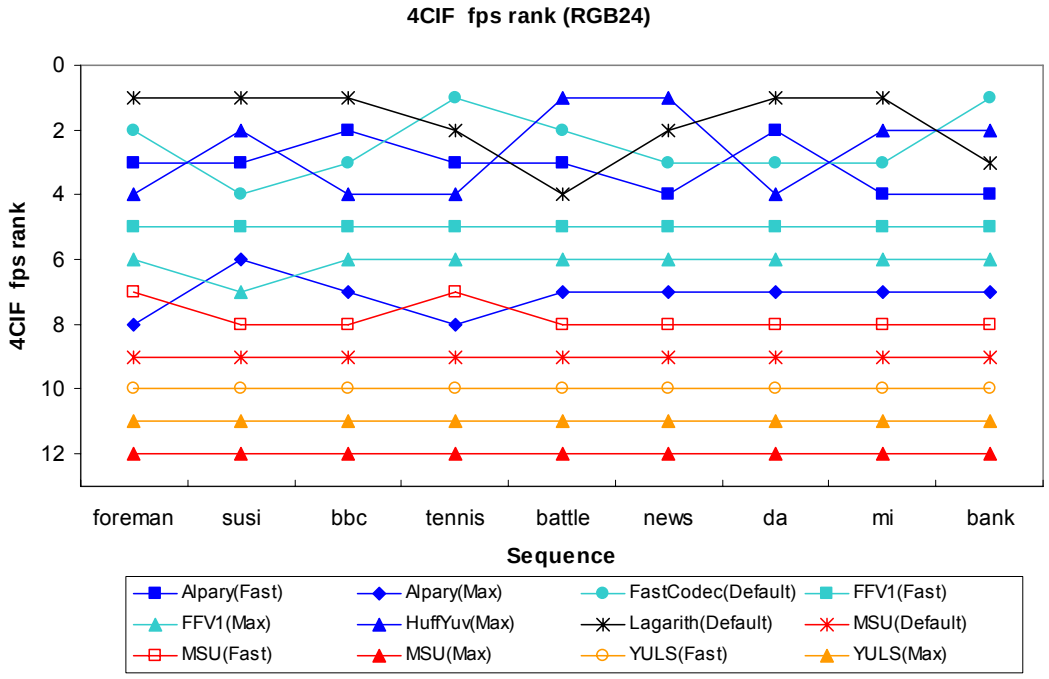


Figure 81. Maximum Compression: Per-file 4CIF fps ranking for RGB24

http://www.compression.ru/video/

4.4.2 YUY2

Here results are noticeably different from those of RGB24. **YULS** is a **definite leader** being 5-6% better than FFV1 and MSU. Also it should be noted that FFV1's advantage over MSU is not so obvious for YUY2, MSU "Max" preset providing 0.4% better compression than FFV1 "Max", but 157 times (!) slower.

LOCO shows results worse than competitors by both criteria.

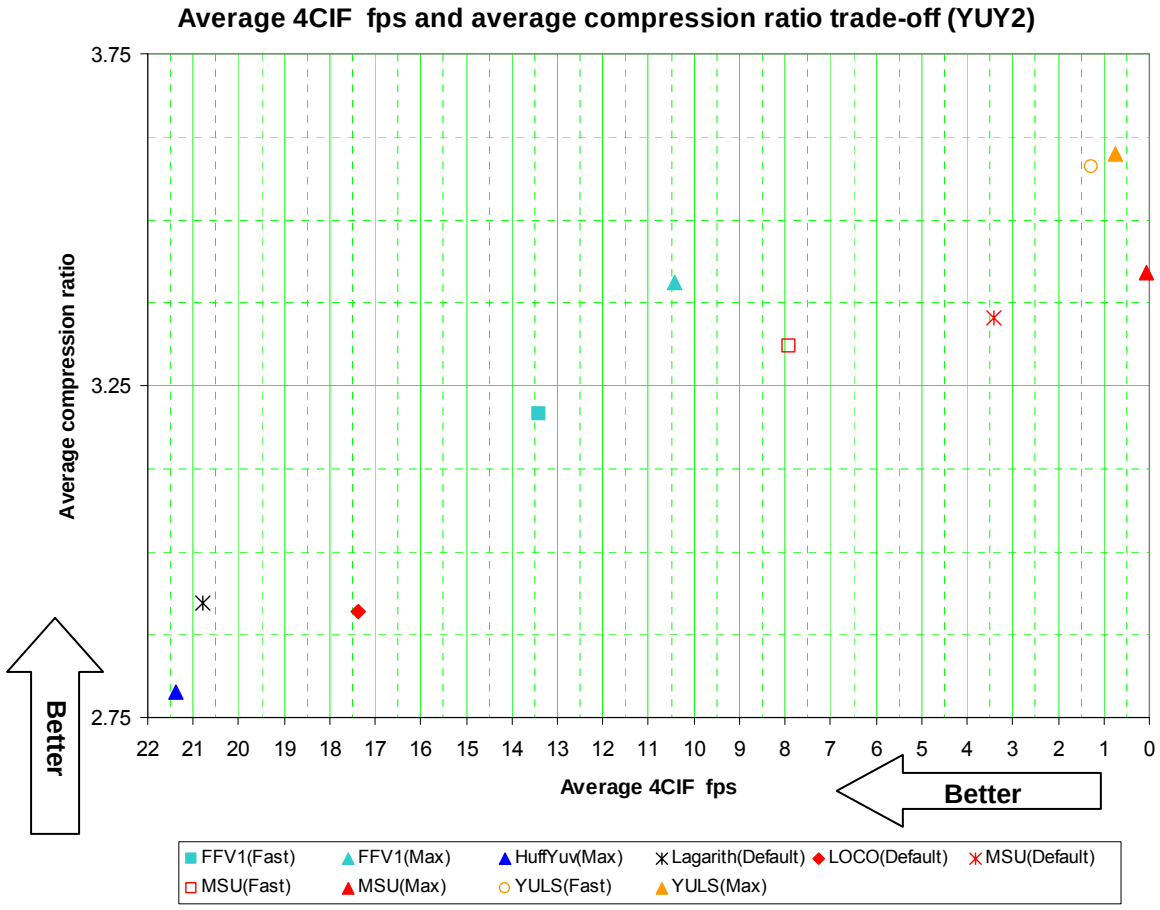


Figure 82. Maximum Compression: Average 4CIF fps and average compression ratio trade-off for YUY2

So recommendations are to use:

- YULS for the best compression;
- FFV1 for medium compression ratio with quite high encoding speed;
- Lagarith to get moderate compression with high speed.

Compression ratios only are showed on Figure 83.

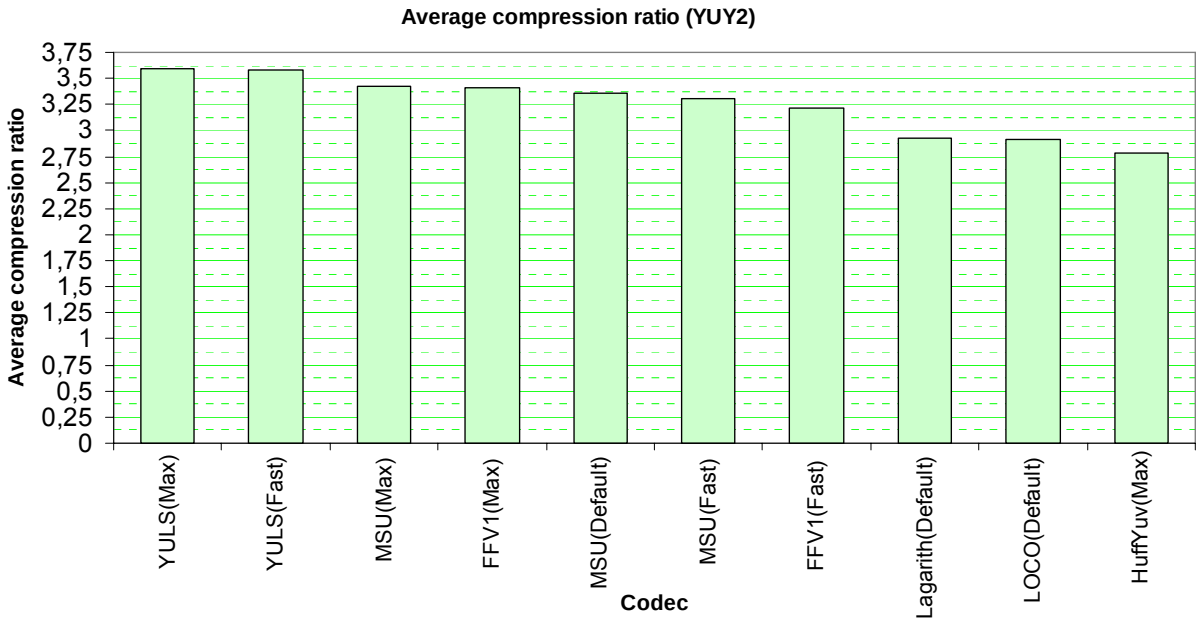


Figure 83. Maximum Compression: Average compression ratio for YUY2

As it can be noted from Figure 84, the leads of YULS over all competitors, of MSU “Max” over FFV1 “Max”, of FFV1 “Max” over MSU “Default” are quite consistent for the entire test set.

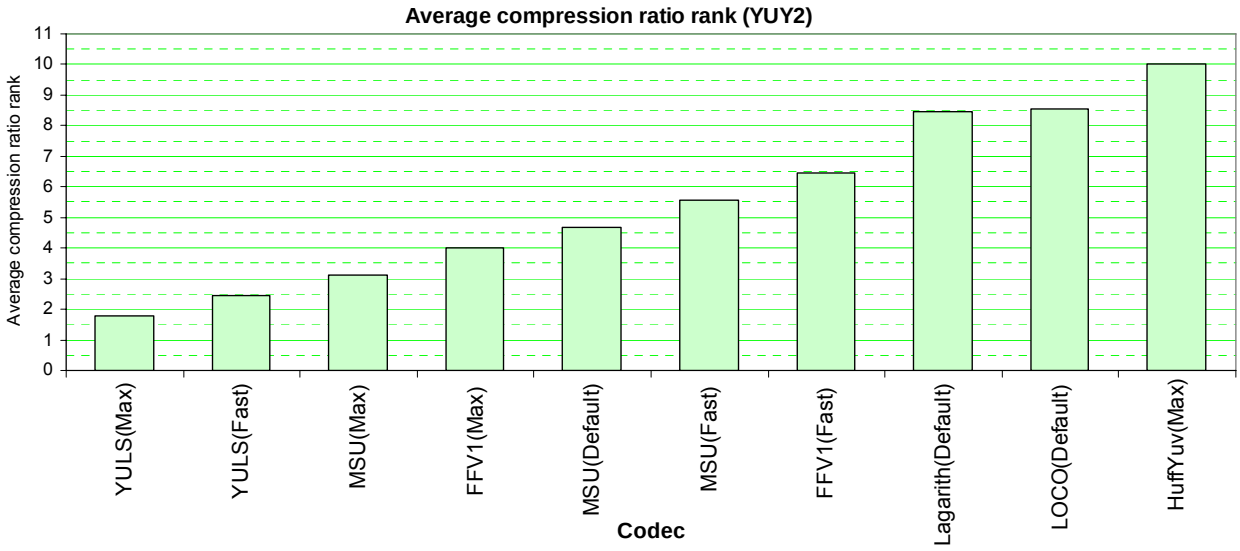


Figure 84. Maximum Compression: Average compression rank for YUY2

Encoding speed results on Figure 85 - Figure 86 are also obvious.

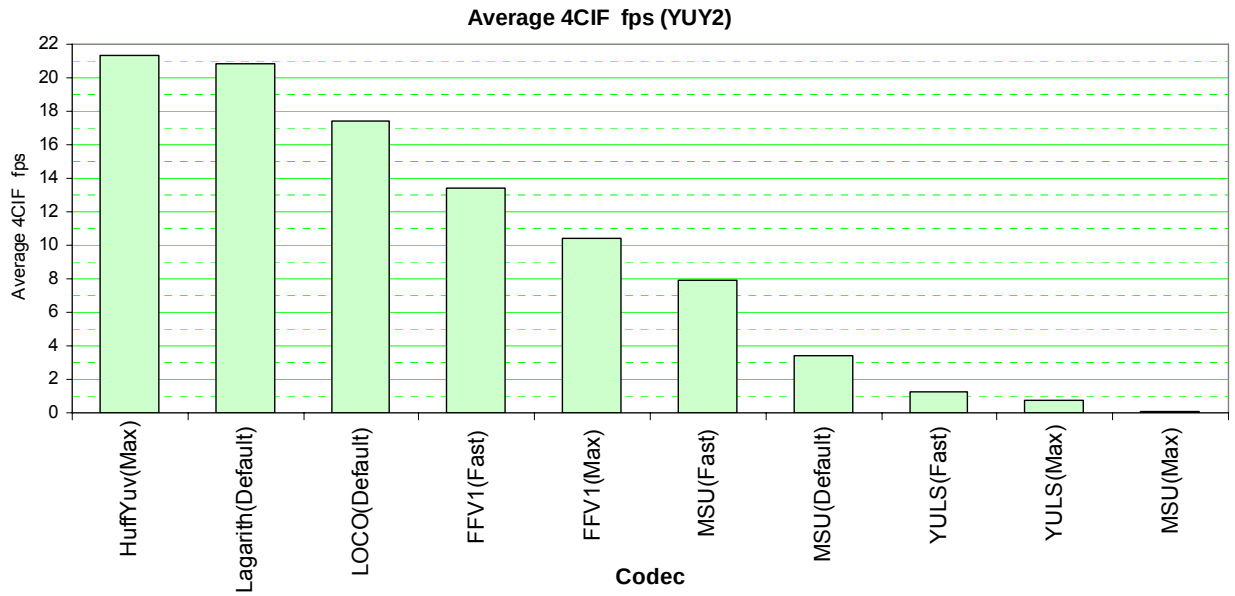


Figure 85. Maximum Compression: Average 4CIF fps for YUY2

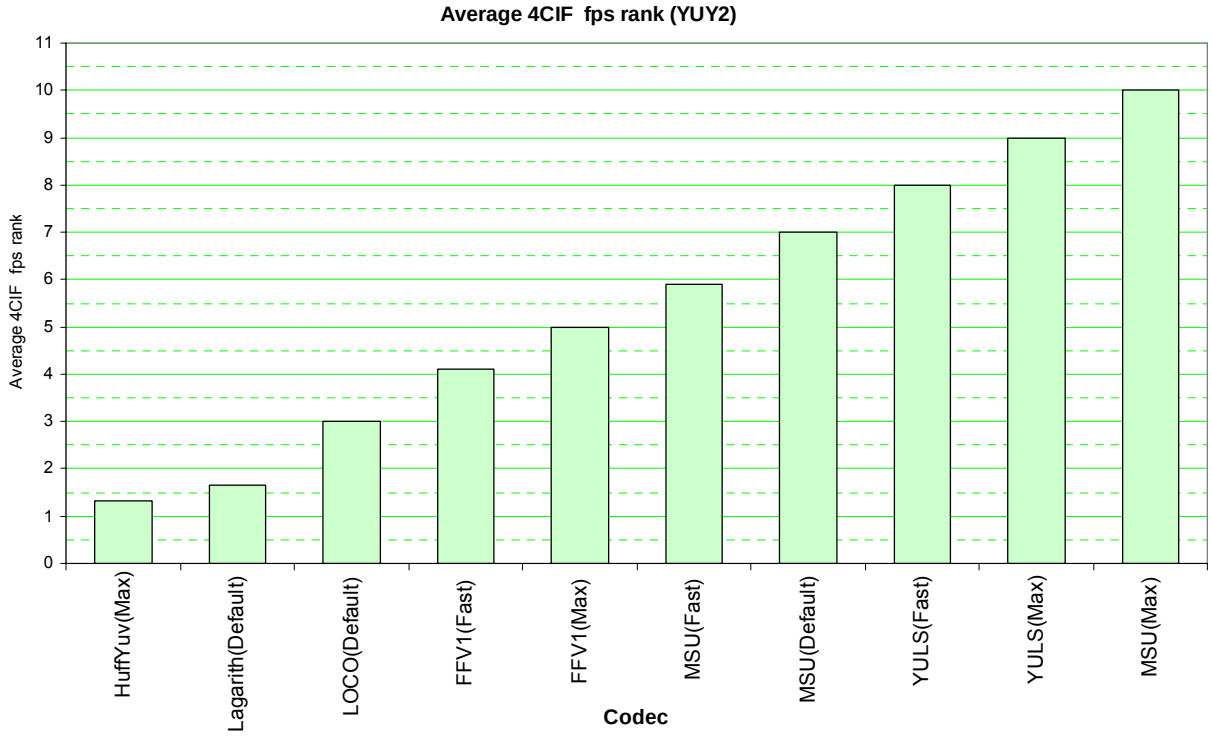


Figure 86. Maximum Compression: Average 4CIF fps rank for YUY2

Figure 87 - Figure 88 demonstrates interesting data on per-file codecs behavior. The main components of YULS' advantage are very good results for highly compressable "MI" and small resolution "Foreman". Also it can be noted than "Tennis" presents a problem for FFV1.

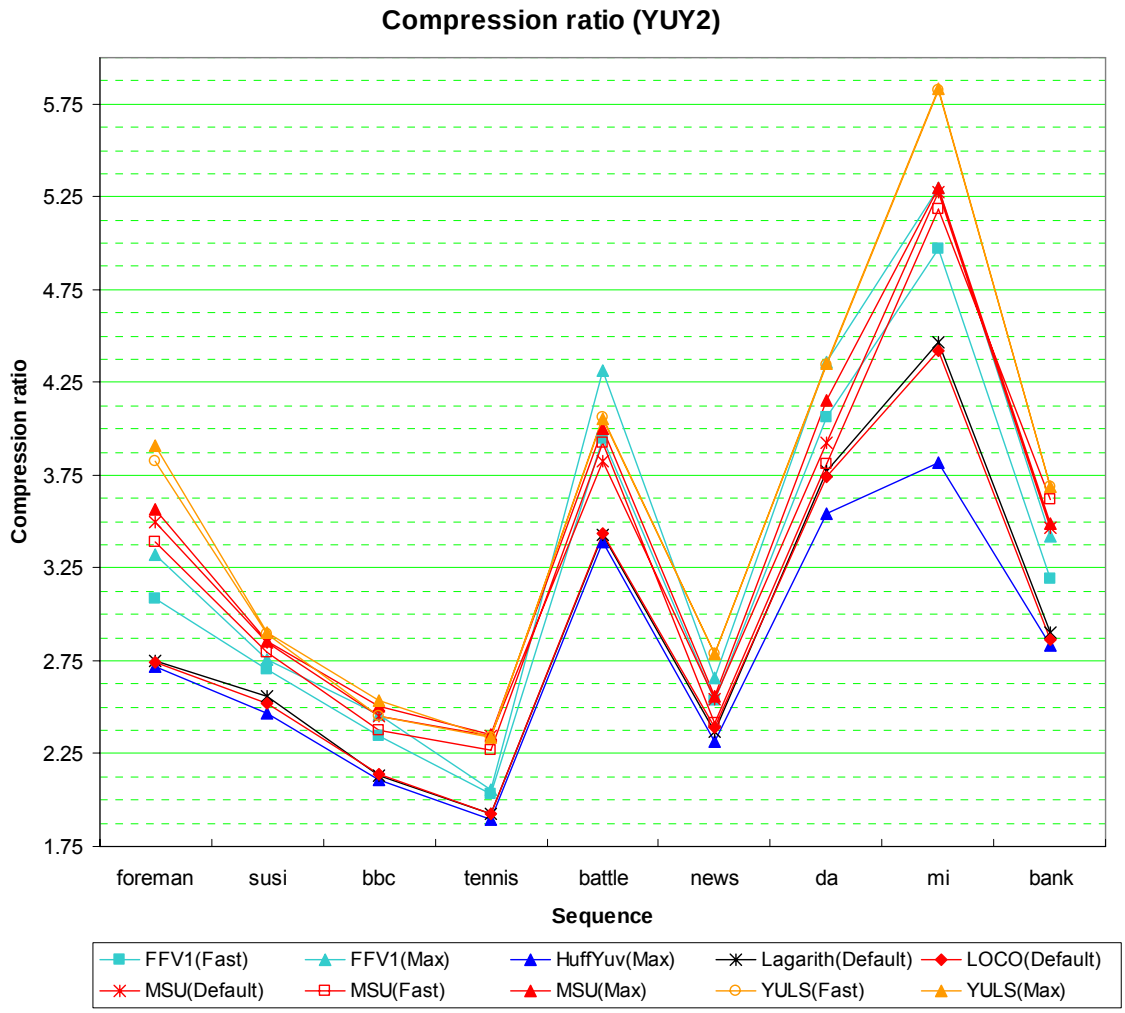


Figure 87. Maximum Compression: Compression ratio for each sequence in YUY2

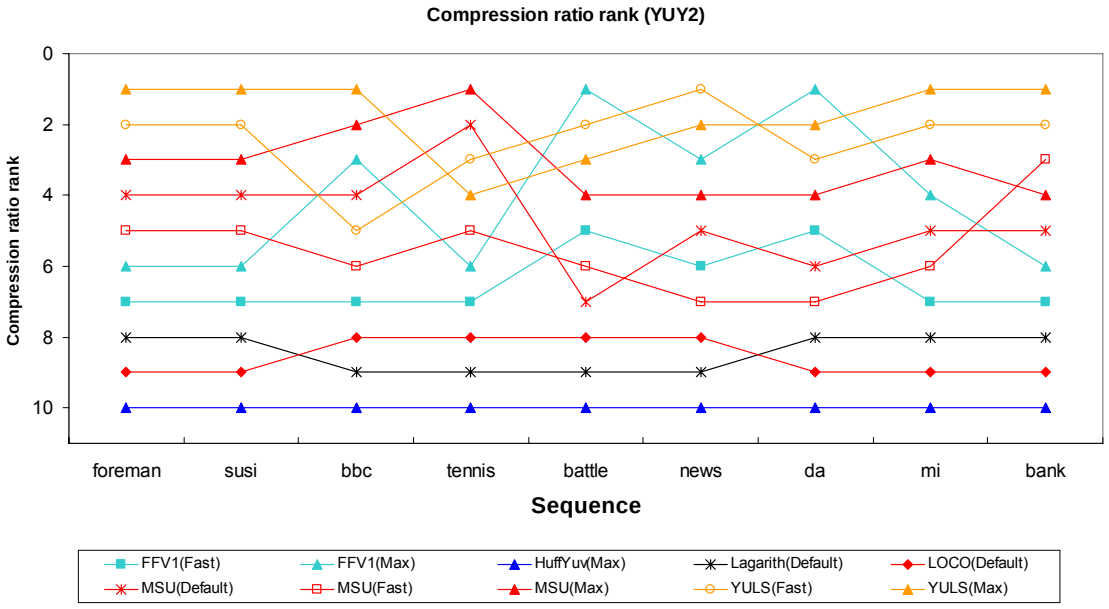


Figure 88. Maximum Compression: Per-file compression ratio ranking for YUY2

http://www.compression.ru/video/

The difference in compression ratio between reference HuffYuv and complex codecs is not so high for YUY2 as for RGB24 due to HuffYUV ffdshow being more complex than Huffyuv 2.1.1 used for RGB24 and, probably, due to less amount of highly compressible chroma data, but this difference gap can achieve more than 1.5 times being 29% on average for our test set (comparing with YULS “Max”).

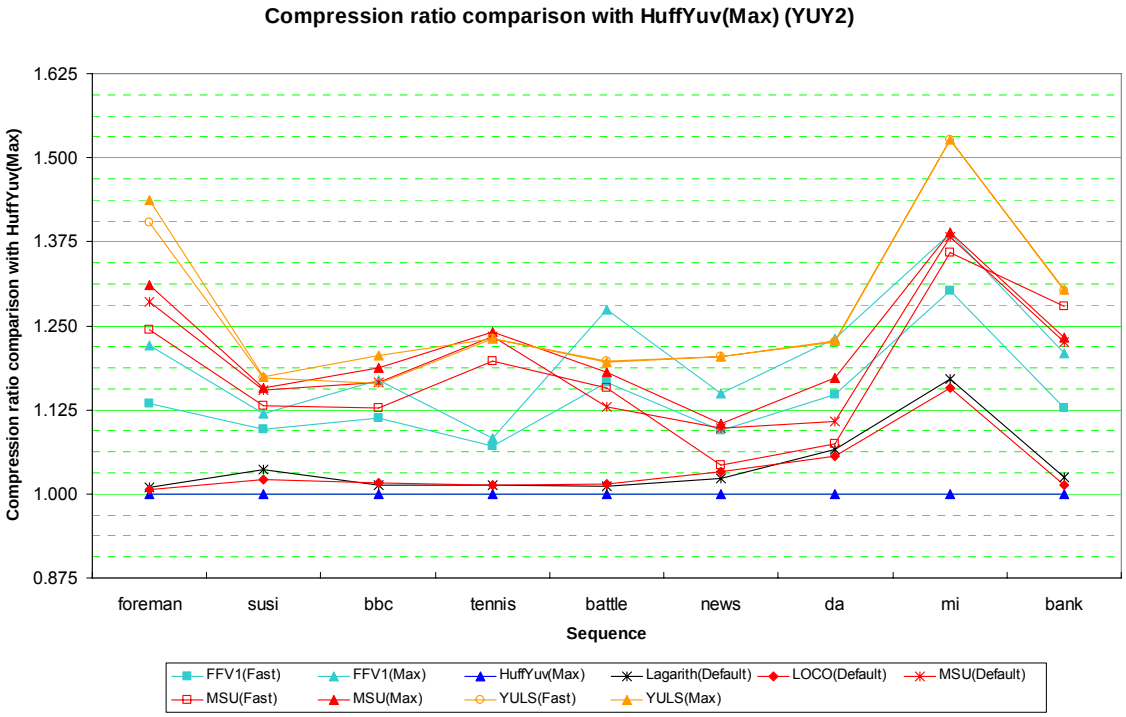


Figure 89. Maximum Compression: Relative per-file compession ratio for YUY2 in comparison to HuffYuv

Per-file speed data do not contain any surprises (Figure 90 - Figure 91).

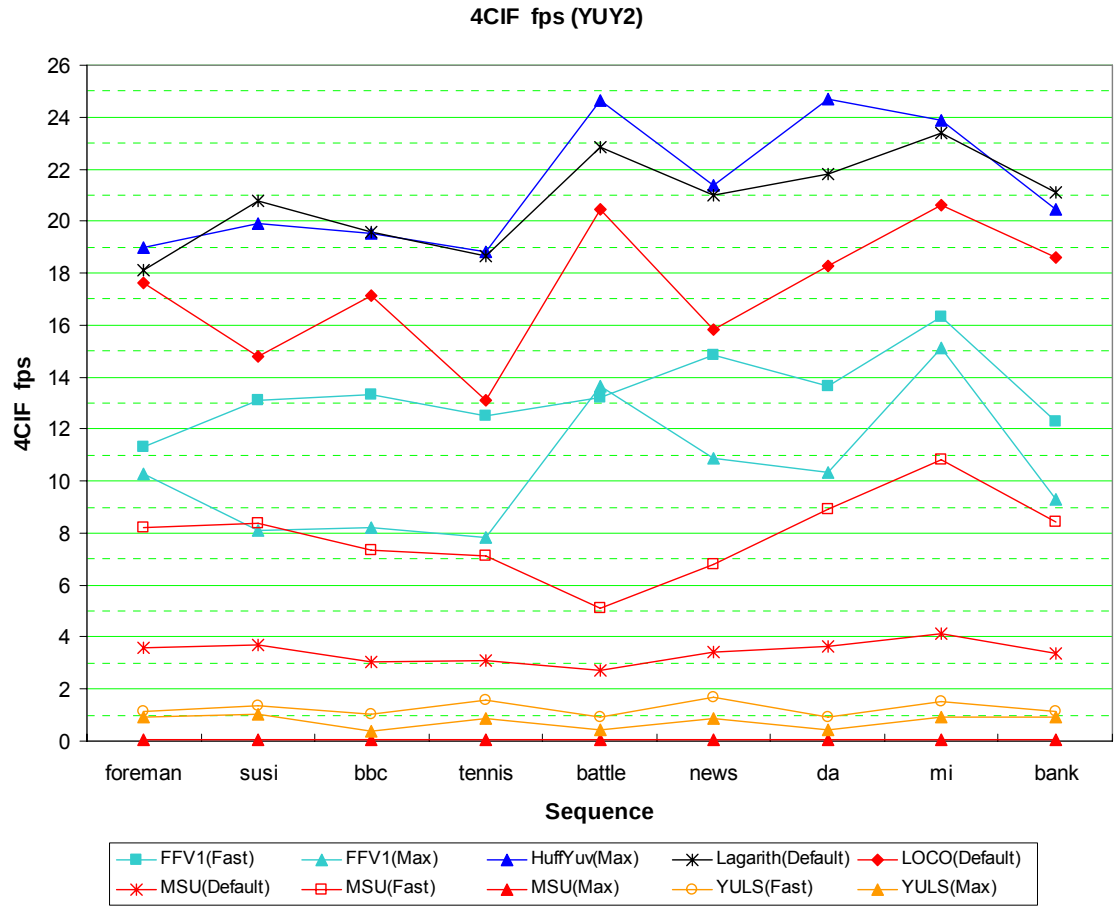


Figure 90. Maximum Compression: Per-file 4CIF fps for YUY2

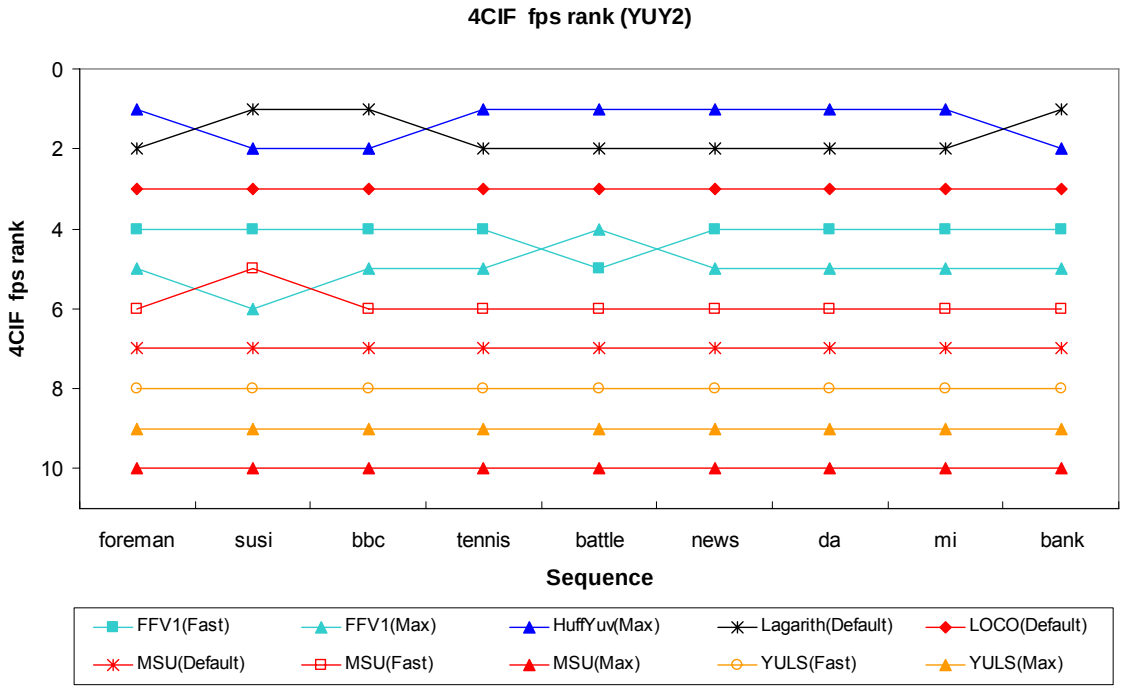


Figure 91. Maximum Compression: Per-file 4CIF fps ranking for YUY2

http://www.compression.ru/video/

4.4.3 YV12

YV12 results resemble YUY2 results in general (Figure 92):

- **YULS** is a clear **leader** with 4% of advantage over MSU as the closest competitor.

MSU gives better compression in all modes than FFV1. Also FFV1 and MSU surpass Snow and x264 lossless by both compression ratio and speed. LOCO provides slightly higher compression ratio than Lagarith, but it is substantially slower, so its use can not be recommended.

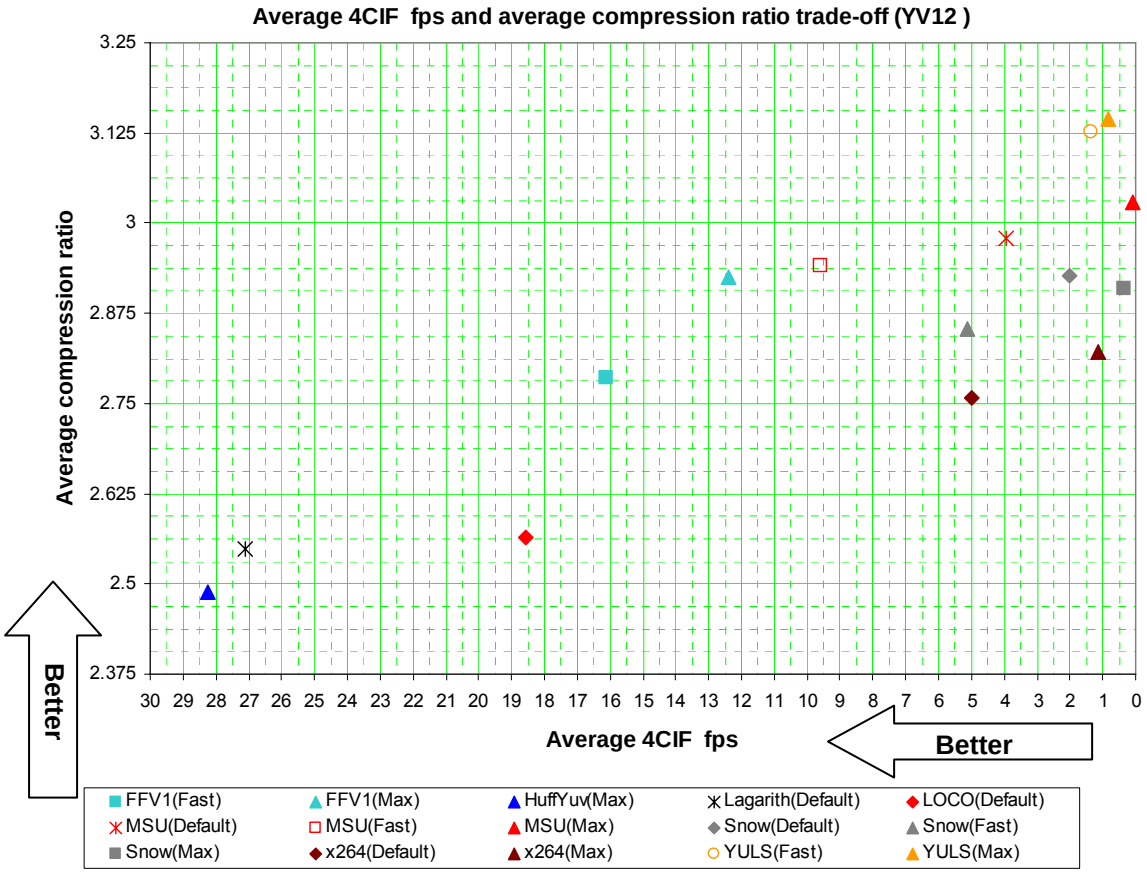


Figure 92. Maximum Compression: Average 4CIF fps and average compression ratio trade-off for YY12

So, it can be concluded that the following compressors are preferable:

- YULS to obtain the best compression;
- MSU “Default” and MSU “Fast” to get medium compression in a faster way;
- FFV1 to obtain relatively good compression with high speed.

Figure 93 - Figure 94 highlights compression ratio only results. Compressors generally show stable results over entire test set, as Figure 94 illustrates.

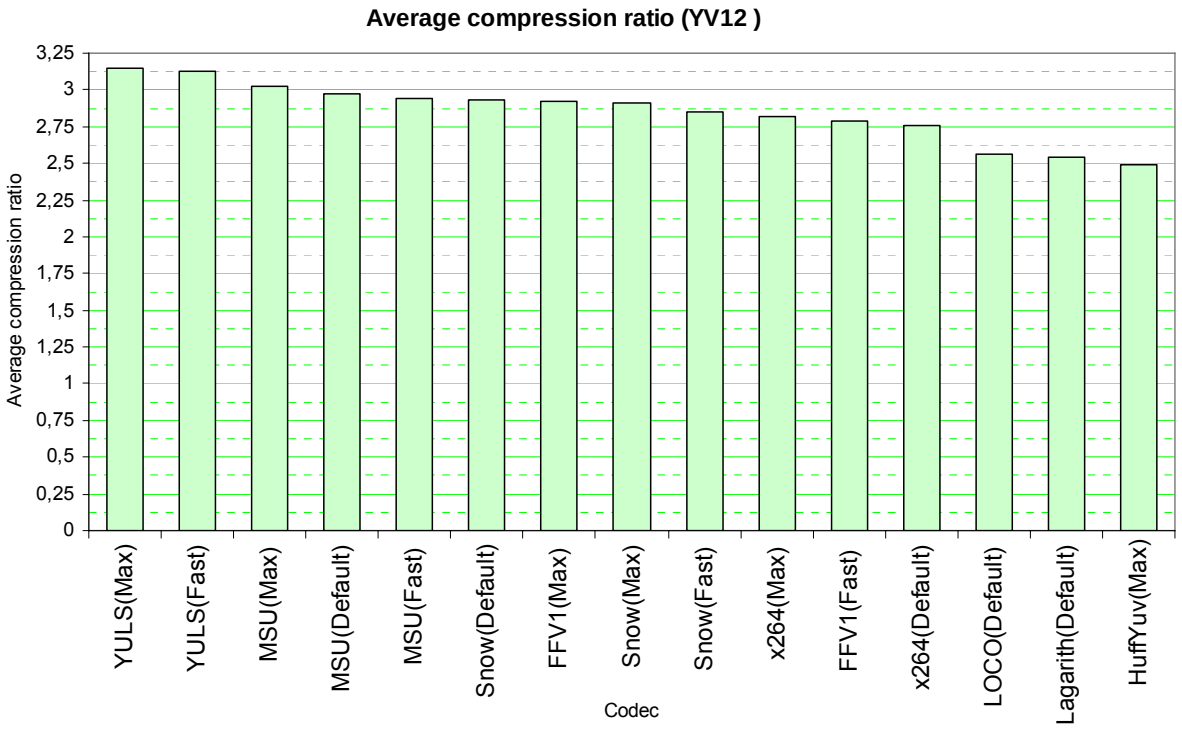


Figure 93. Maximum Compression: Average compression ratio for YV12

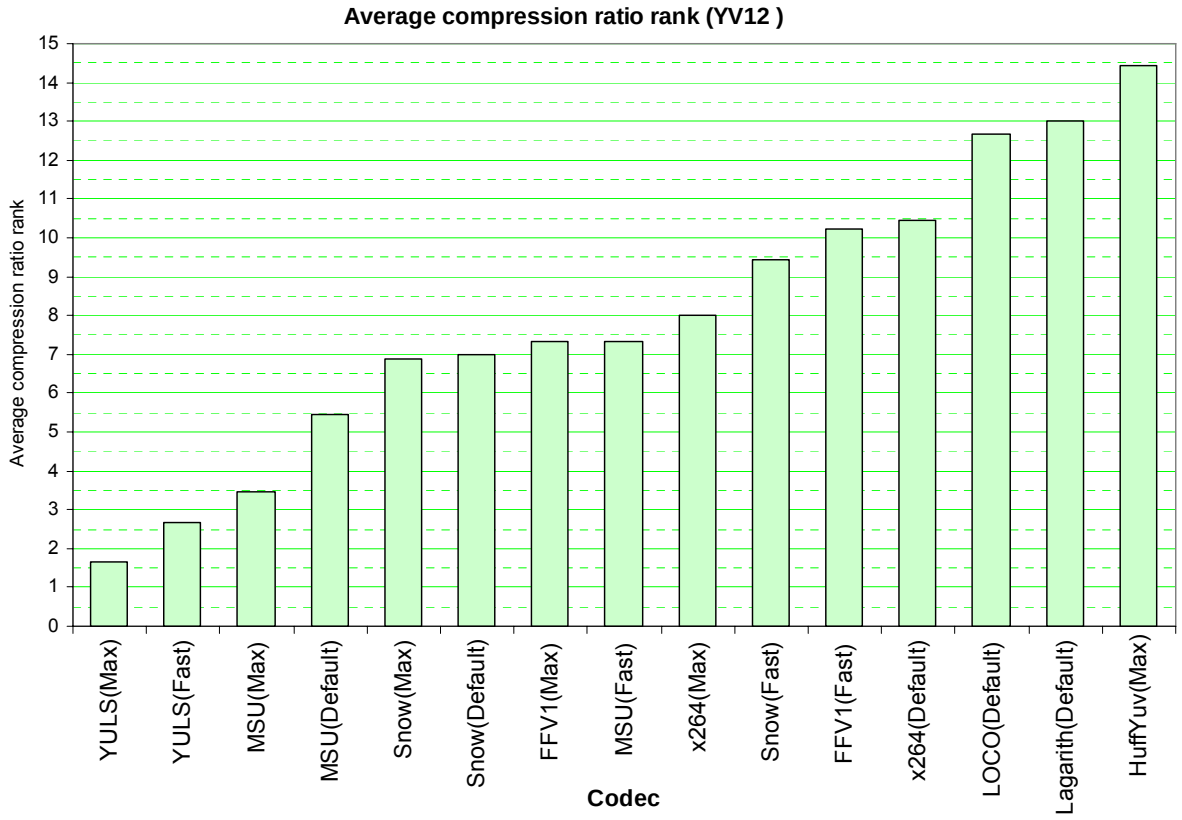


Figure 94. Maximum Compression: Average compression rank for YV12

From Figure 95 - Figure 96 it can be concluded that Snow “Max” and MSU “Max” are hardly to be used in practice, and that per-file performance are stable for the tested codecs.

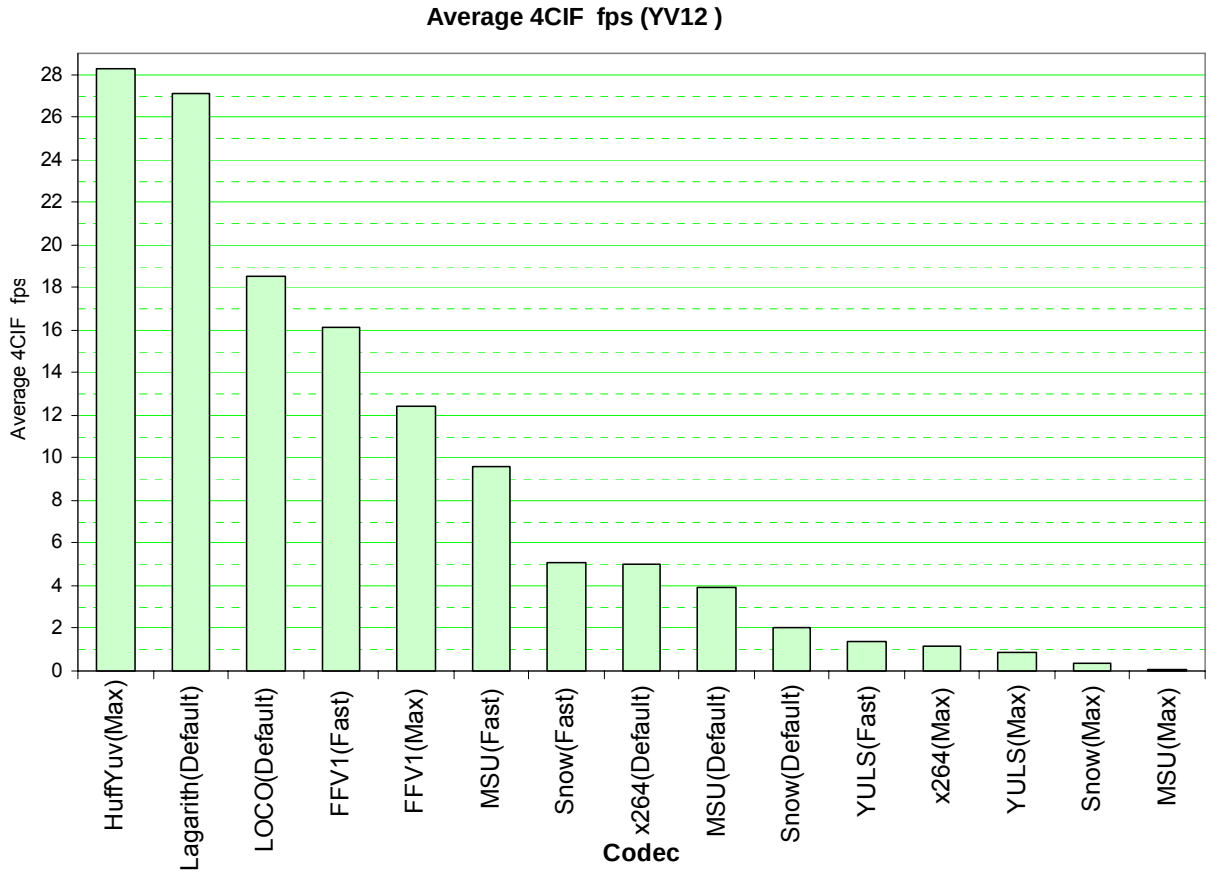


Figure 95. Maximum Compression: Average 4CIF fps for YV12

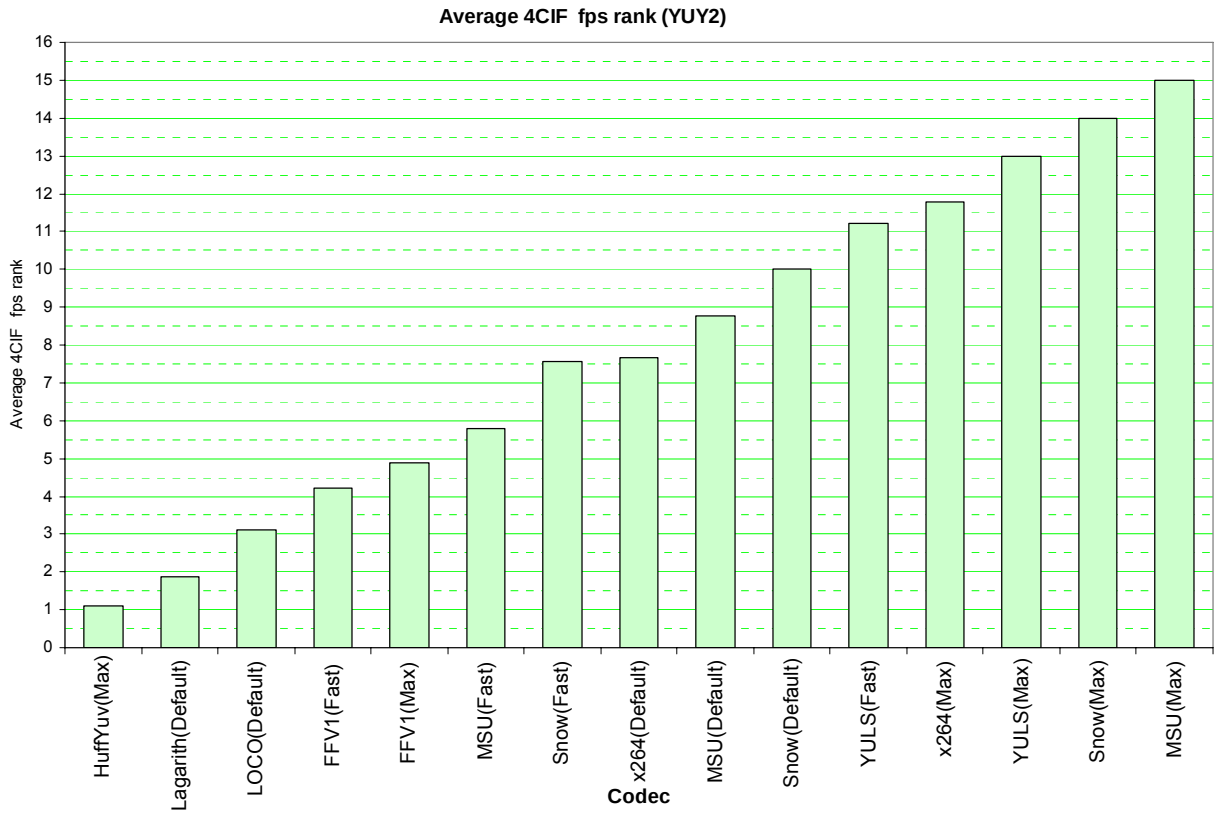


Figure 96. Maximum Compression: Average 4CIF fps rank for YV12

Per-file compression ratio results (Figure 97 - Figure 99) are similar to those of YUY2. YULS considerably surpass other codecs on “Foreman”, “DA”, “MI”. In contrast to RGB24, it provides clearly the best compression on ‘News”. As for other color spaces, FFV1 is the compression ratio leader on “Battle”. x264 is an underdog for “Battle”, “News”, and “DA”. The difference between HuffYuv and YULS can achieve 1.4 times being 1.26 times on average.

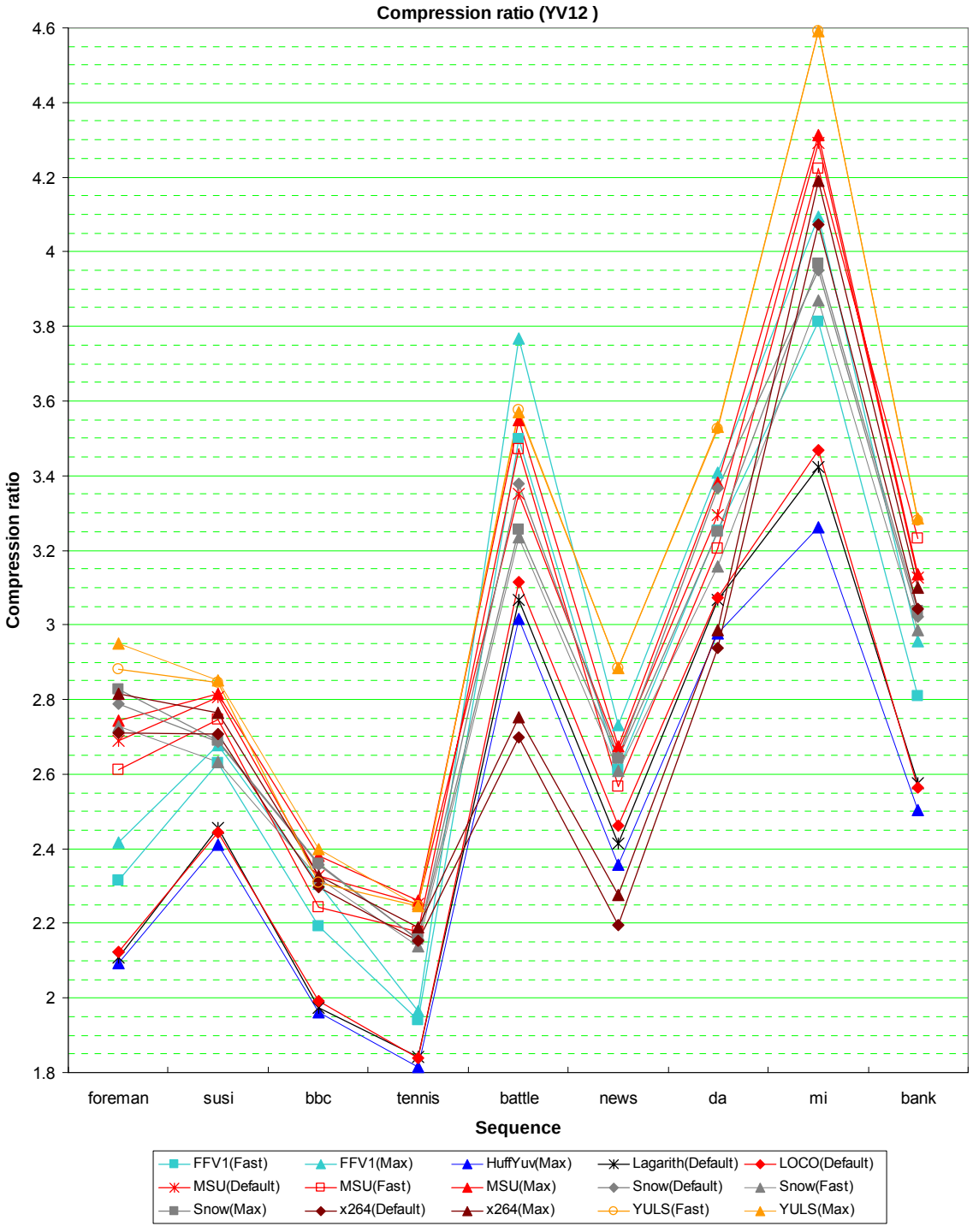


Figure 97. Maximum Compression: Compression ratio for each sequence in YV12

http://www.compression.ru/video/

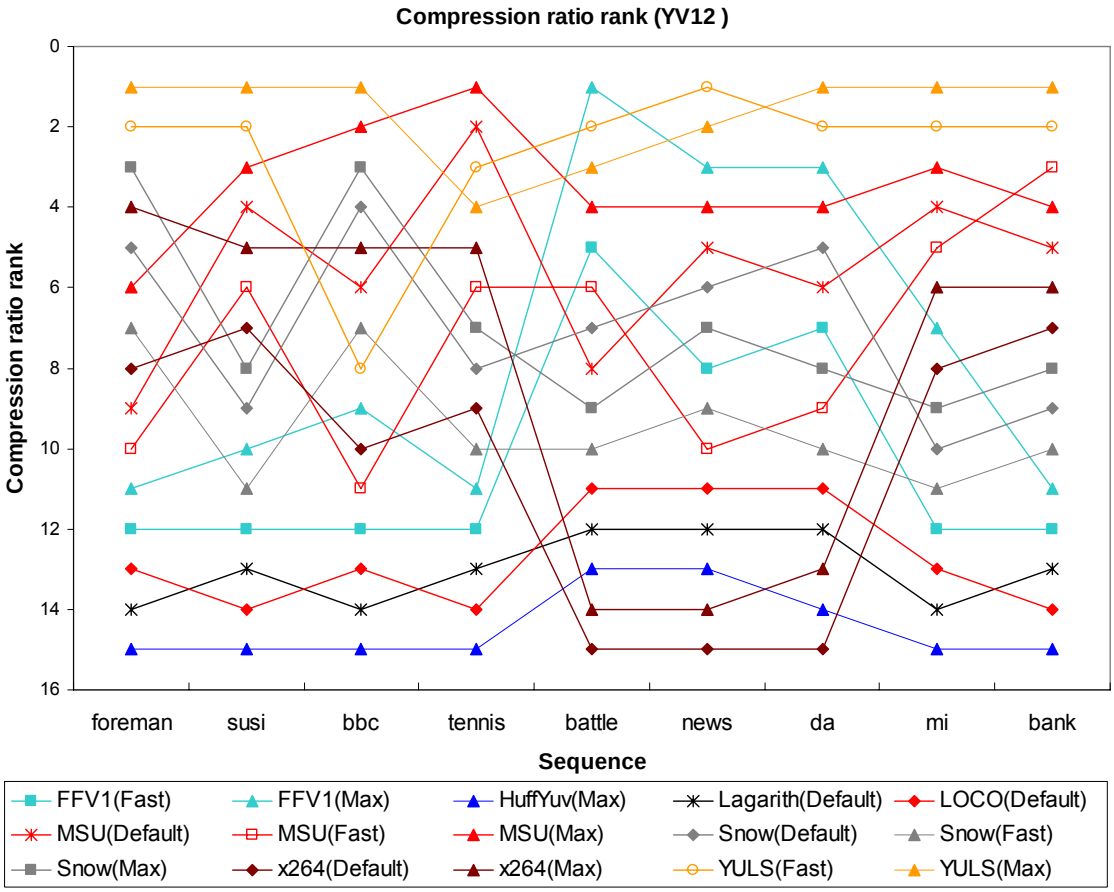


Figure 98. Maximum Compression: Per-file compression ratio ranking for YV12

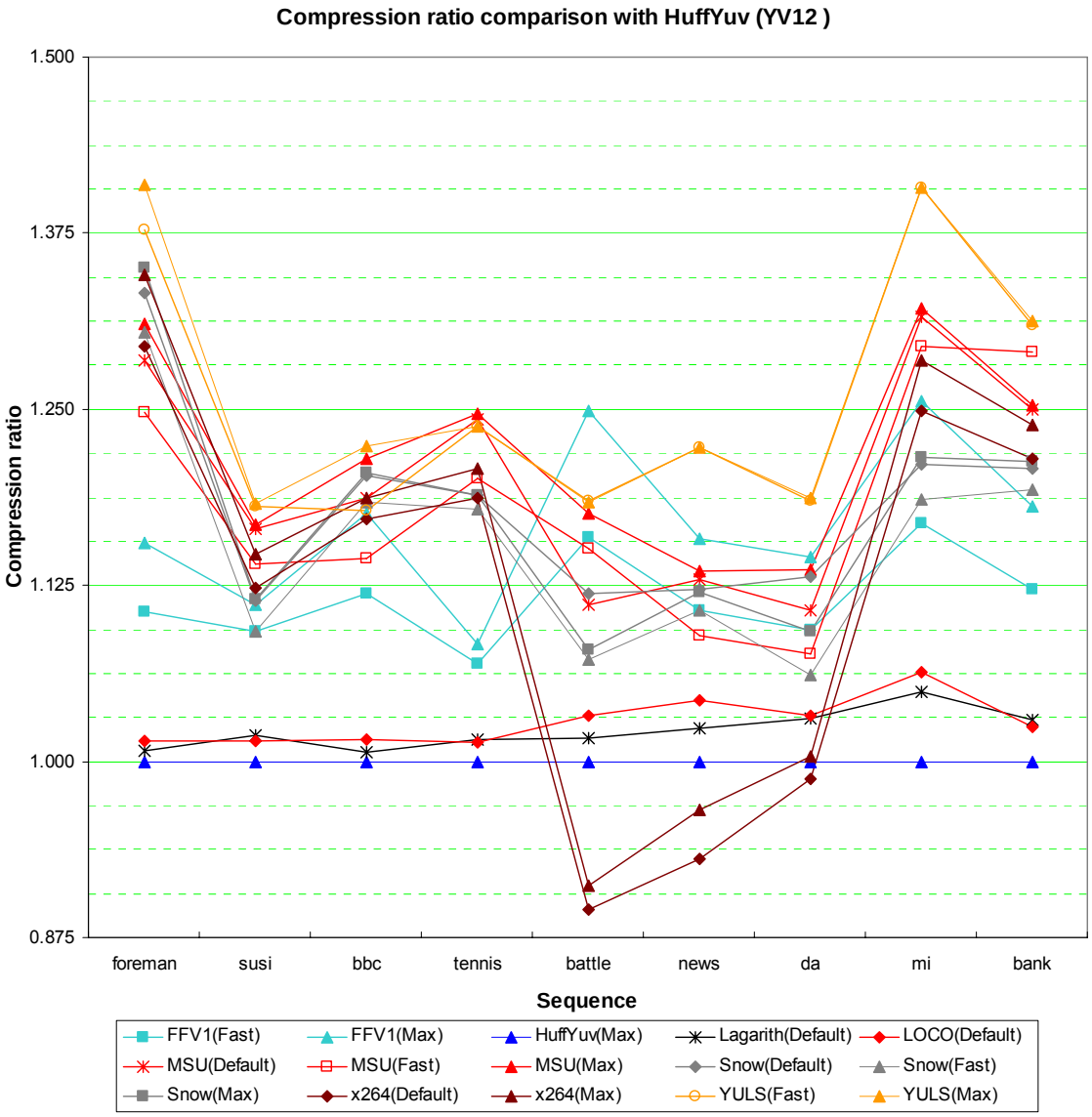


Figure 99. Maximum Compression: Relative per-file compession ratio for YV12 in comparison to HuffYuv

Per-file encoding speed results are predictable on the whole, see Figure 100 - Figure 101. Among relatively fast codecs the most stable performance is provided by FFV1 most probably.

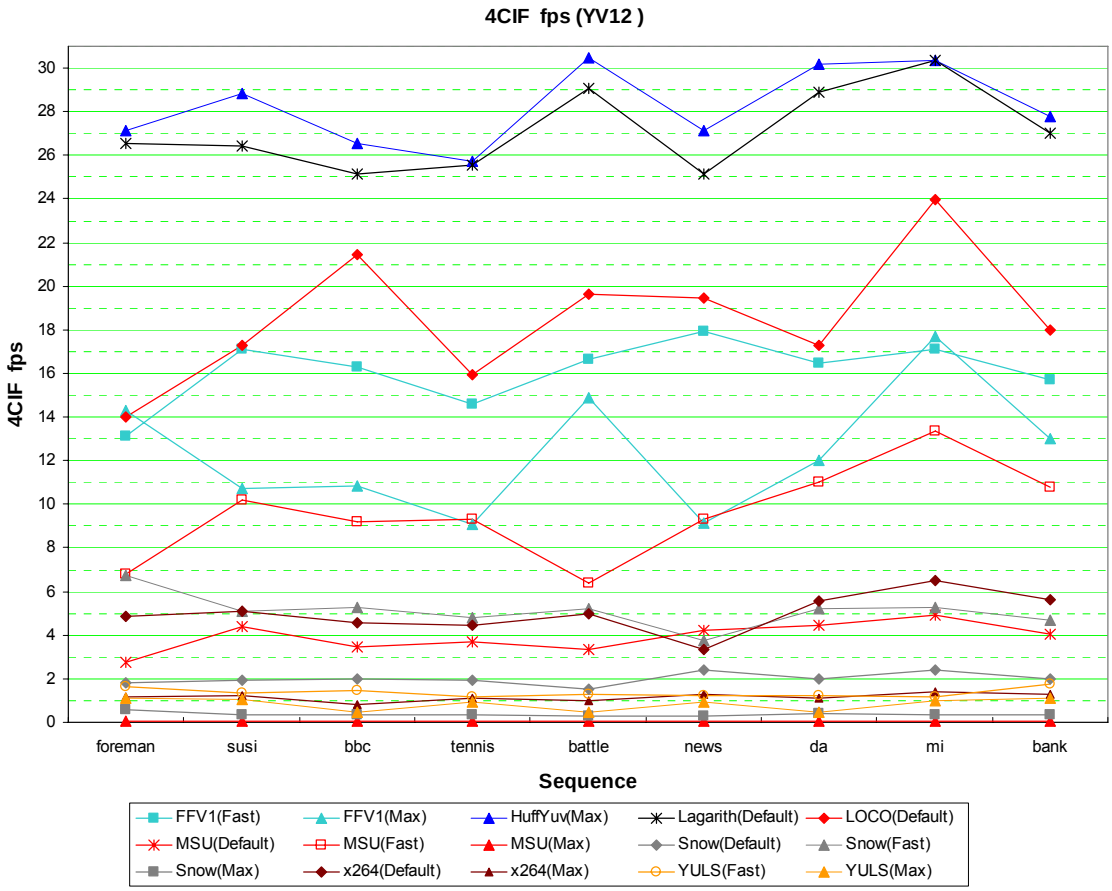


Figure 100. Maximum Compression: Per-file 4CIF fps for YV12

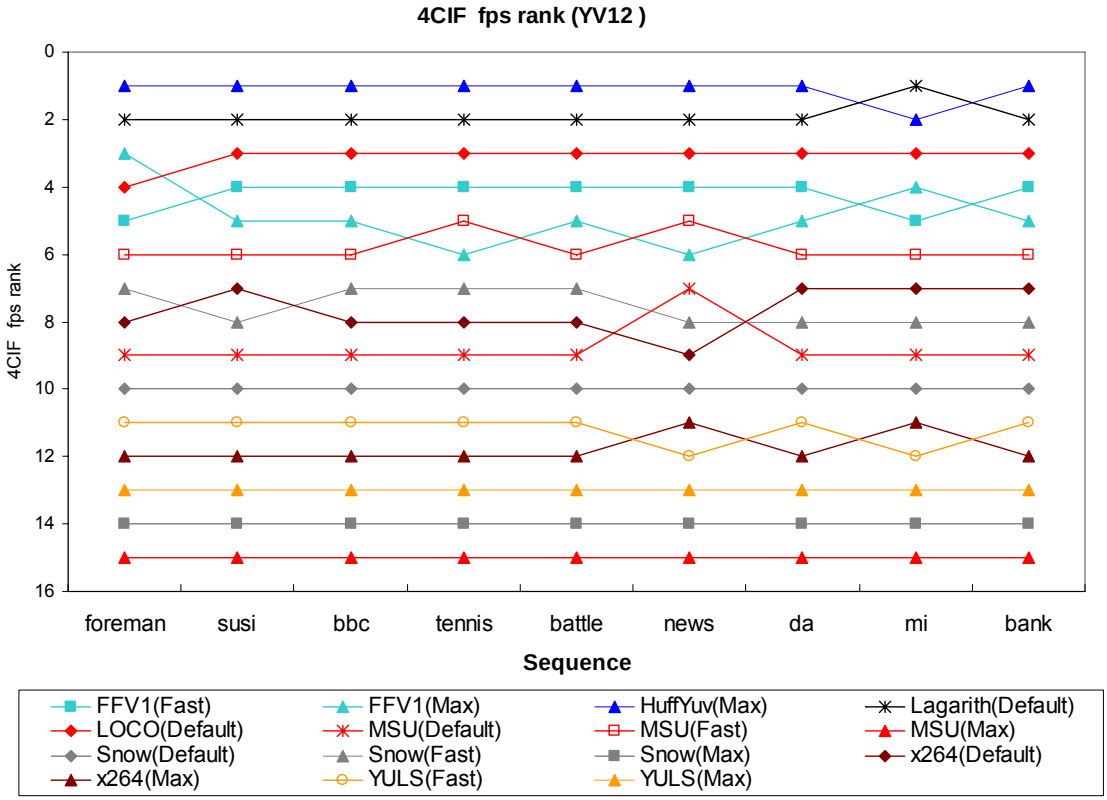


Figure 101. Maximum Compression: Per-file 4CIF fps ranking for YV12

4.4.4 Overall Rating

For those interested in combined performance for all three color spaces overall rating was calculated for the proper codecs and presets. Results are presented on Figure 102 and Figure 103. The leaders are YULS, MSU and FFV1, in this order. Among the latter two FFV1 is likely preferable since MSU “Max” is dramatically slow.

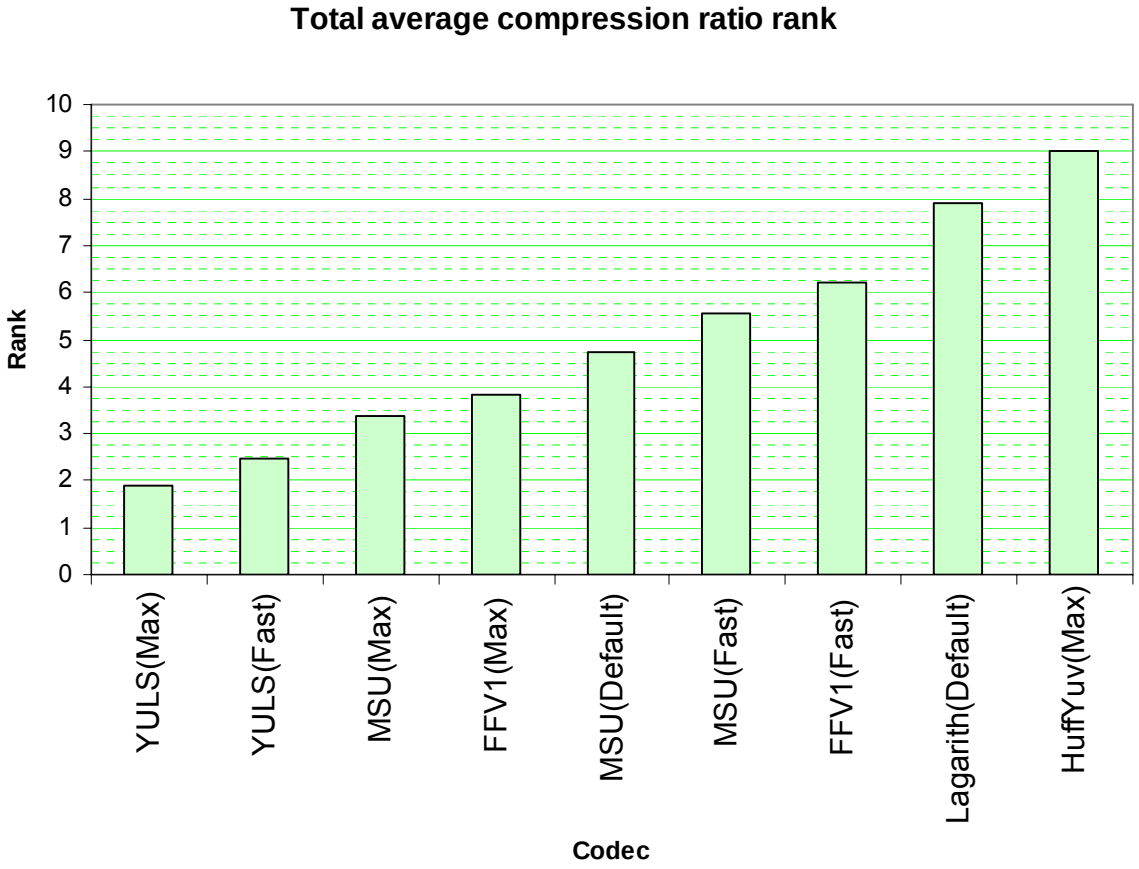


Figure 102. Maximum Compression: Total average compression ratio rank for RGB24, YUY2, YV12

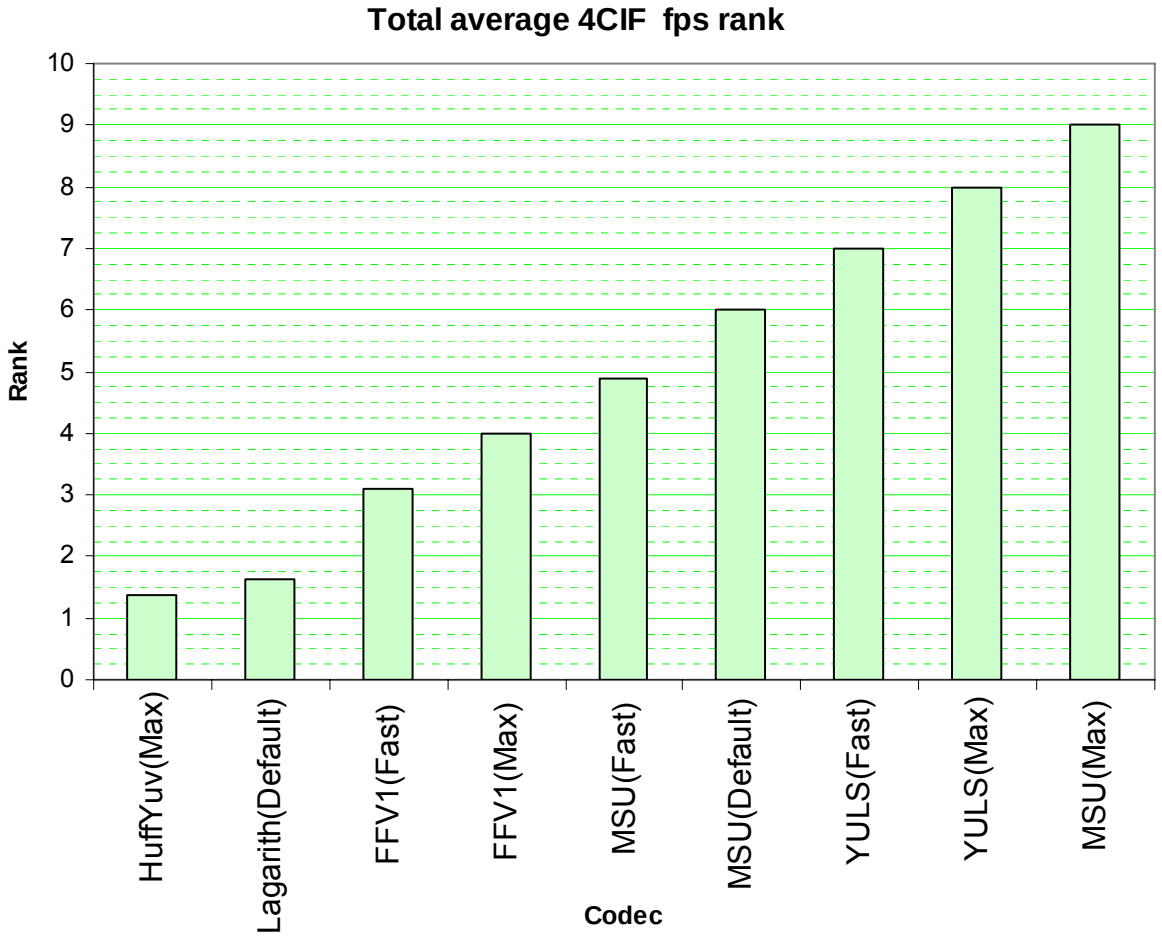


Figure 103. Maximum Compression: Total average 4CIF fps rank for RGB24, YUY2, YV12

4.5 Multithreading Processing Speed

Three codecs from this comparison support multi-threading, namely: AviZlib, Lagarith and x264. In this section a comparison of default modes with multi-threaded ones will be shown. For Lagarith and x264 additional small testing was performed on a server described in “Environment” during which “Battle” sequence was encoded as the longest one in our comparison. The difference in compression ratio between single and multi-threaded modes was smaller than 1% for all codecs, so compression ratio charts will be omitted.

4.5.1 AviZlib

Codec	Available relevant parameters	Presets	
		Fast	Fast MT
AVIzlib	Compress Mode:	Hi Speed	Hi Speed
	Video Format(24bit Only):	RGB24	RGB24
	Multi Tread:	Off	On
	PNG Filter:	Off	Off

http://www.compression.ru/video/

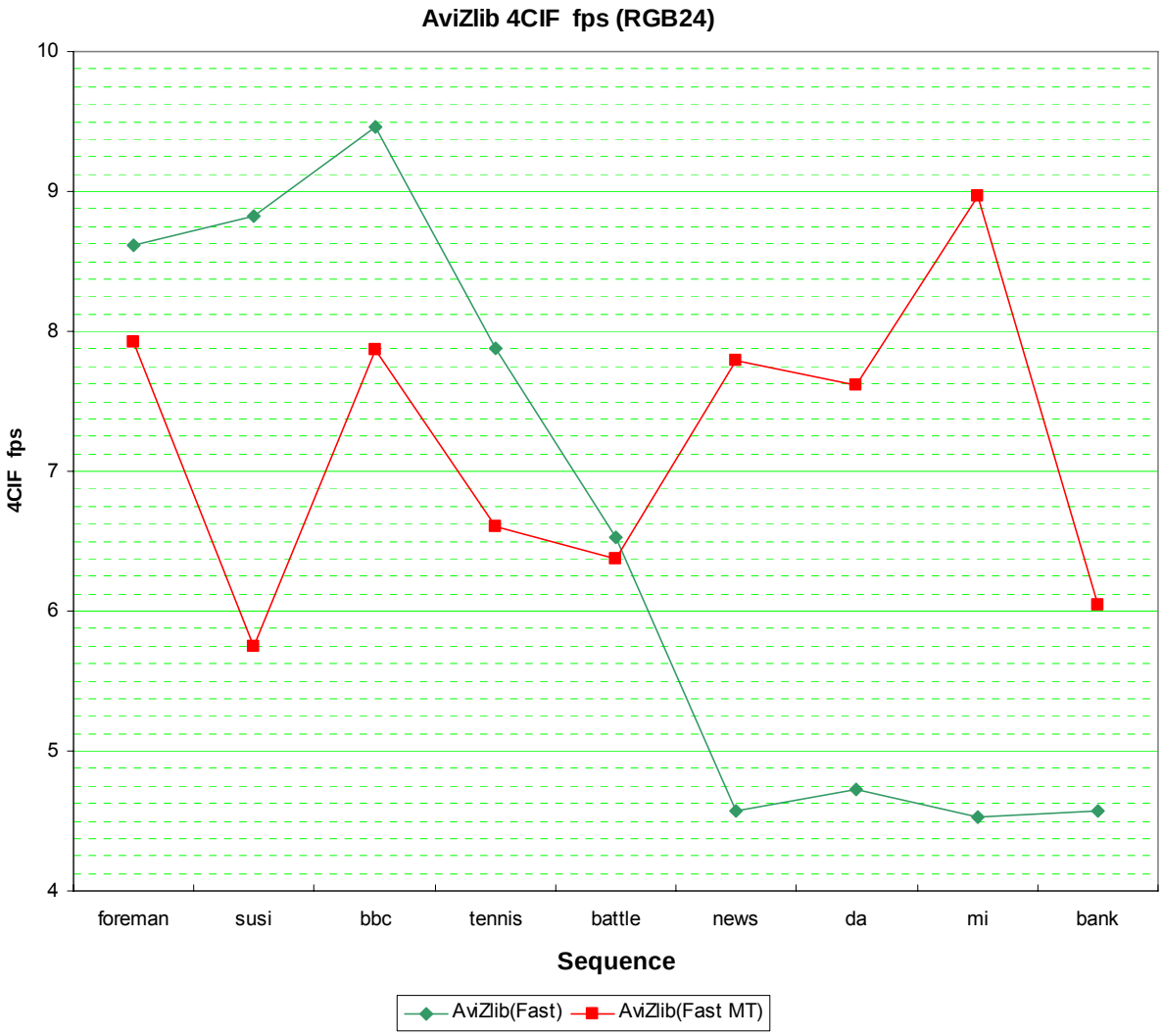


Figure 104. Multithreading: Per-file 4CIF fps for AviZlib, RGB24

http://www.compression.ru/video/

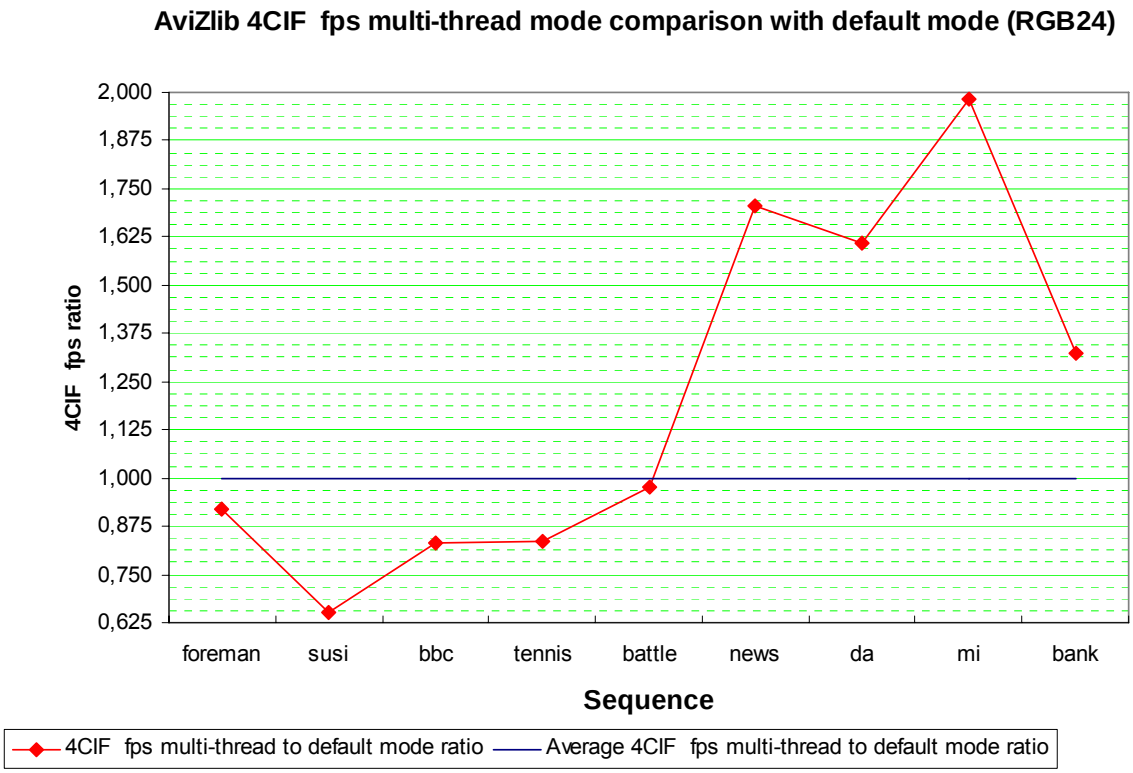


Figure 105. Multithreading: Per-file 4CIF fps for AviZlib in comparison to single-thread mode, RGB24

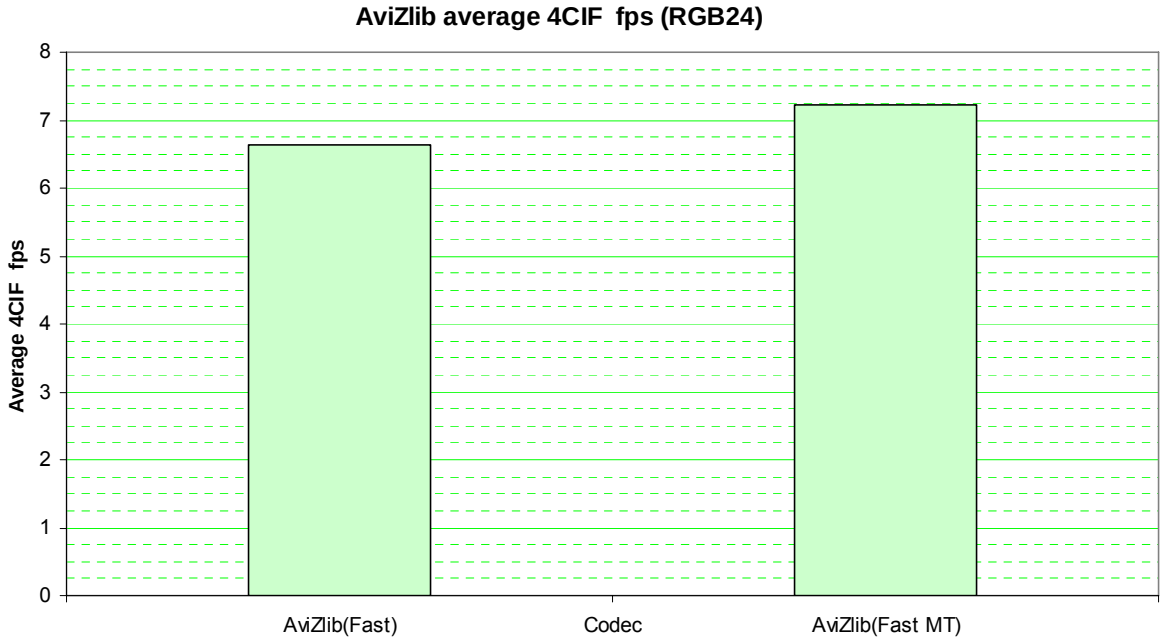


Figure 106. Multithreading: Average 4CIF fps for AviZlib, RGB24

4.5.2 Lagarith

Codec	Available relevant parameters	Presets	
		Default	MT
Lagarith	Enable Null Frames	On	On
	Use multithreading:	Off	On

RGB24

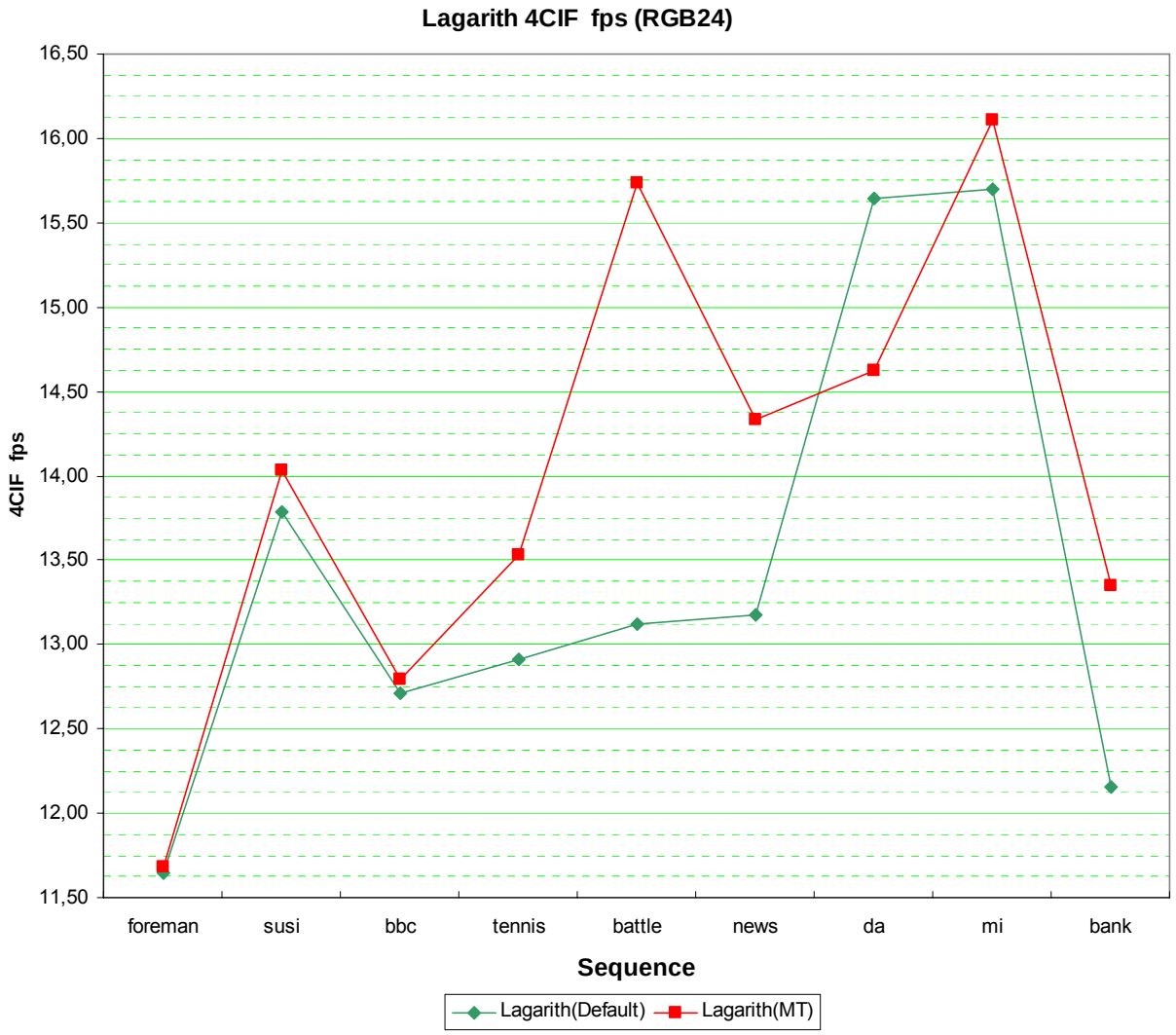


Figure 107. Multithreading: Per-file 4CIF fps for Lagarith, RGB24

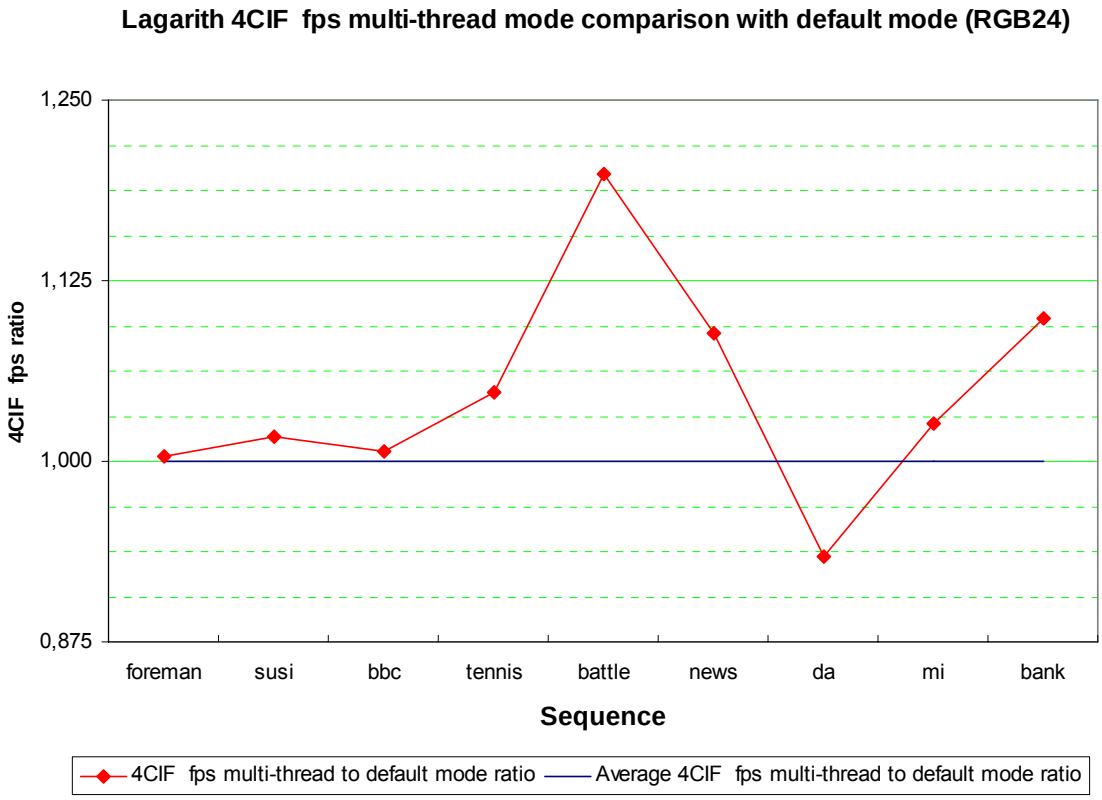


Figure 108. Multithreading: Per-file 4CIF fps for Lagarith in comparison to single-thread mode, RGB24

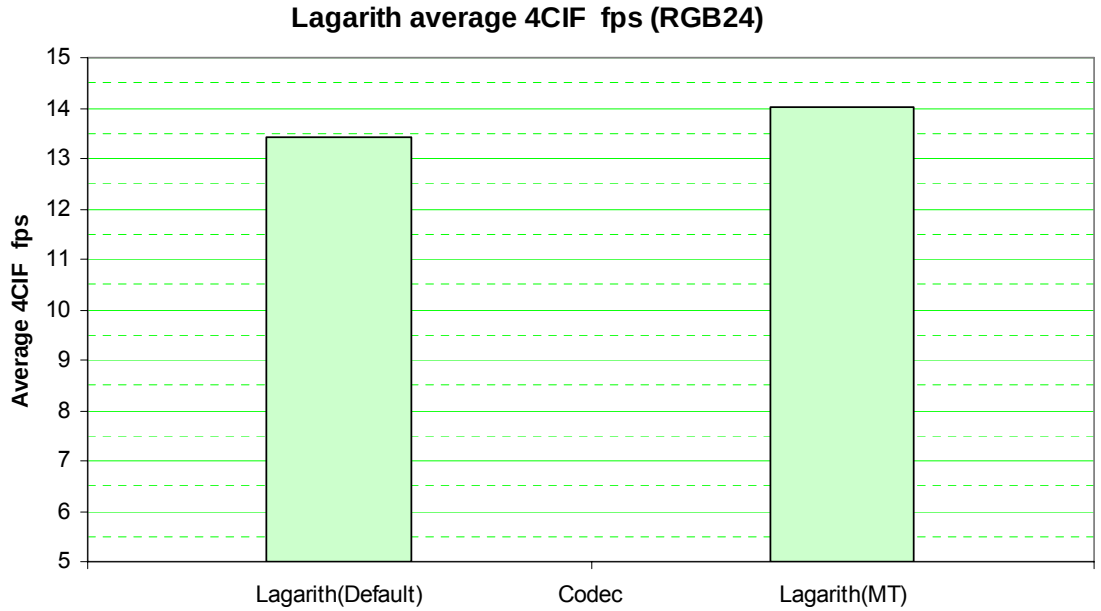


Figure 109. Multithreading: Average 4CIF fps for Lagarith, RGB24

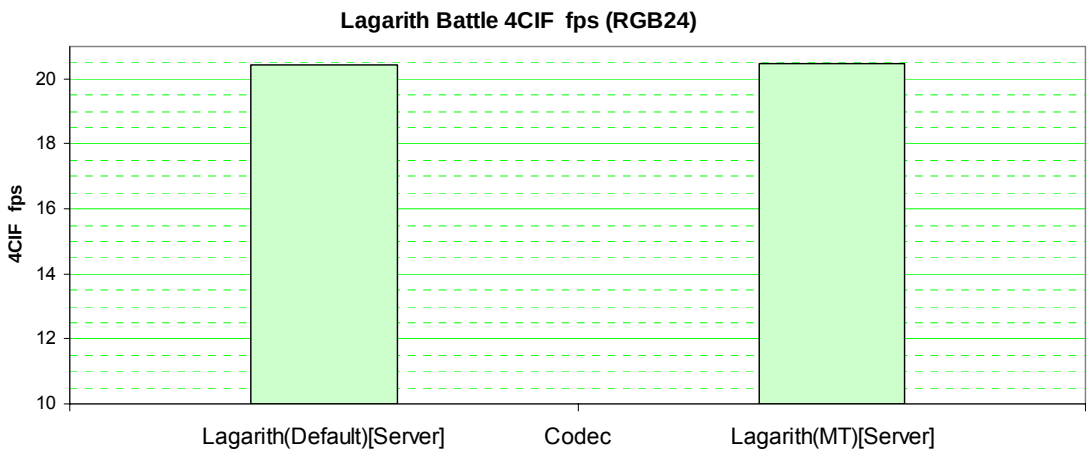


Figure 110. Multithreading: 4CIF fps for Lagarith on "Battle", RGB24

YUY2

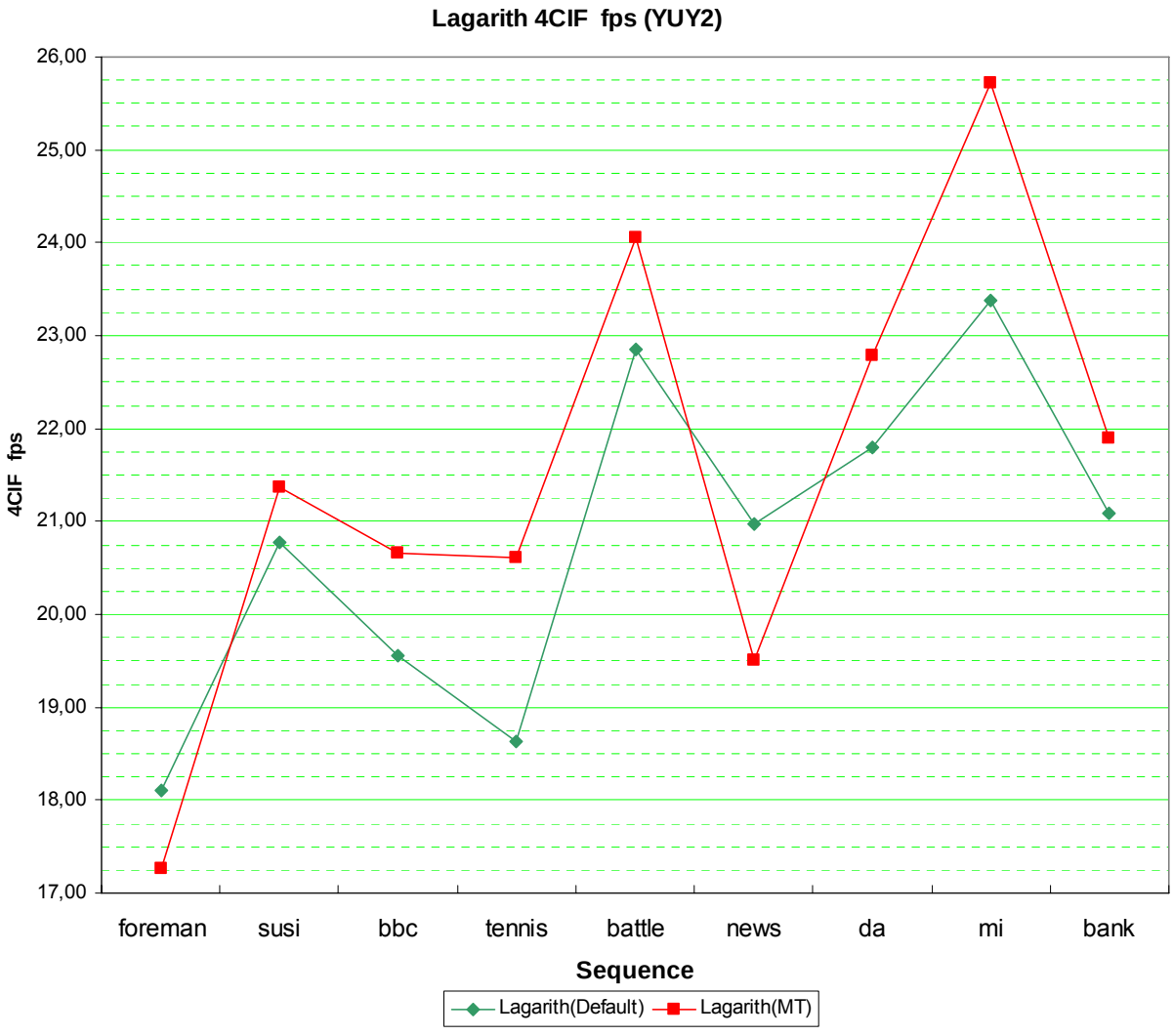


Figure 111. Multithreading: Per-file 4CIF fps for Lagarith, YUY2

http://www.compression.ru/video/

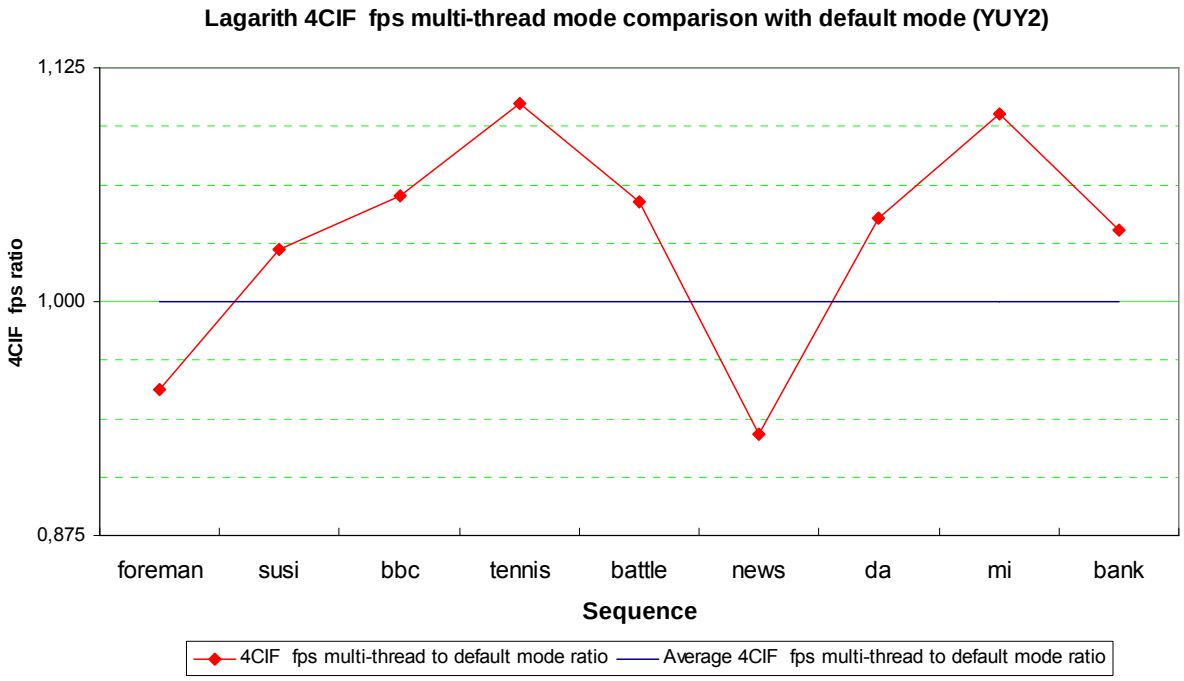


Figure 112. Multithreading: Per-file 4CIF fps for Lagarith in comparison to single-thread mode, YUY2

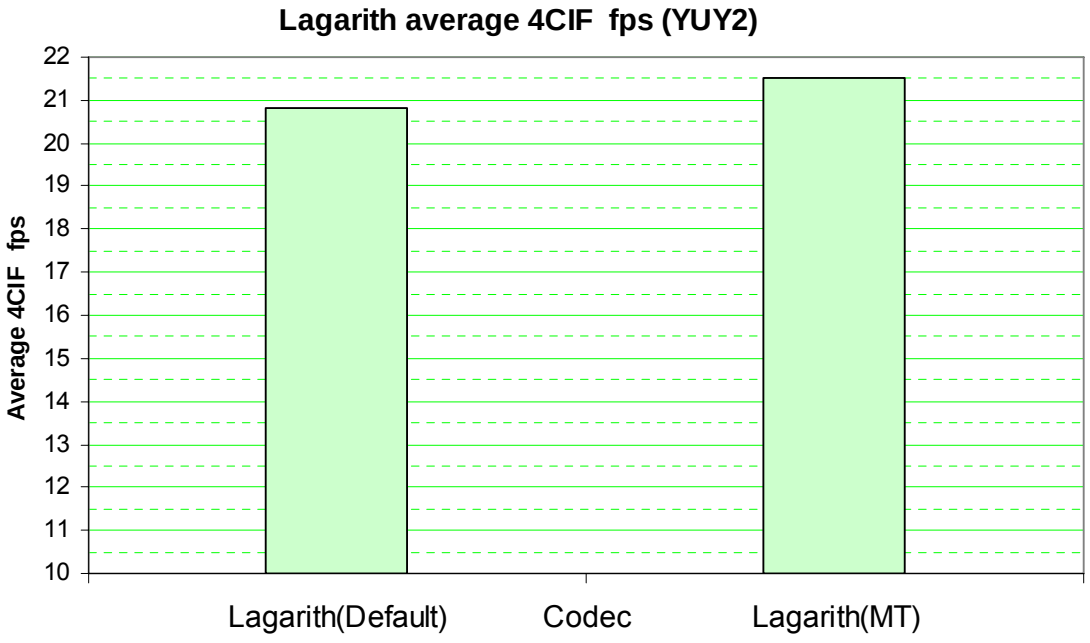


Figure 113. Multithreading: Average 4CIF fps for Lagarith, YUY2

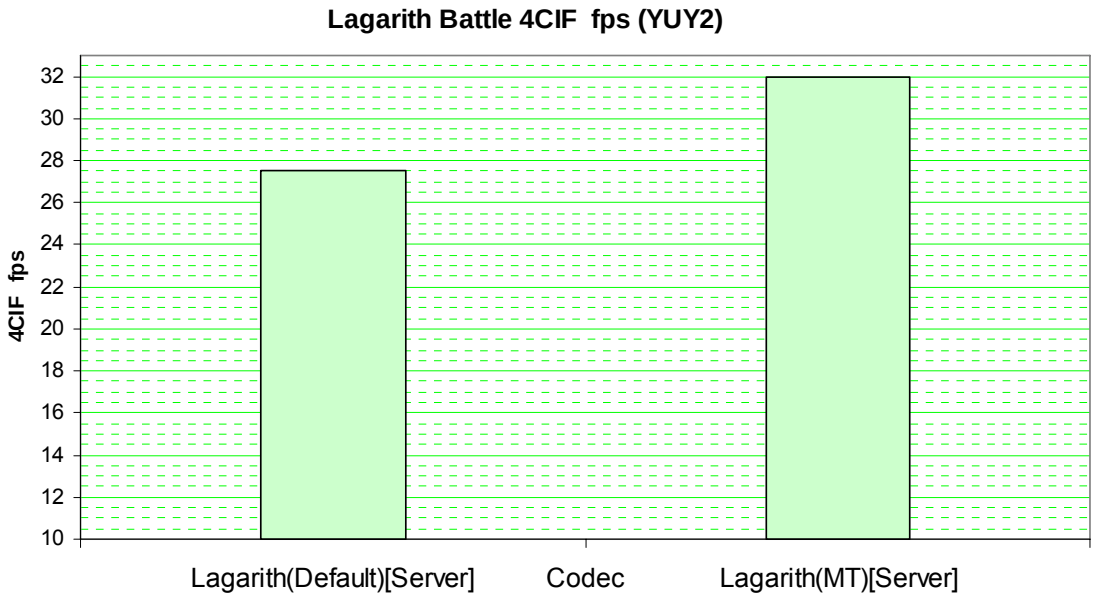


Figure 114. Multithreading: 4CIF fps for Lagarith on "Battle", YUY2

YV12

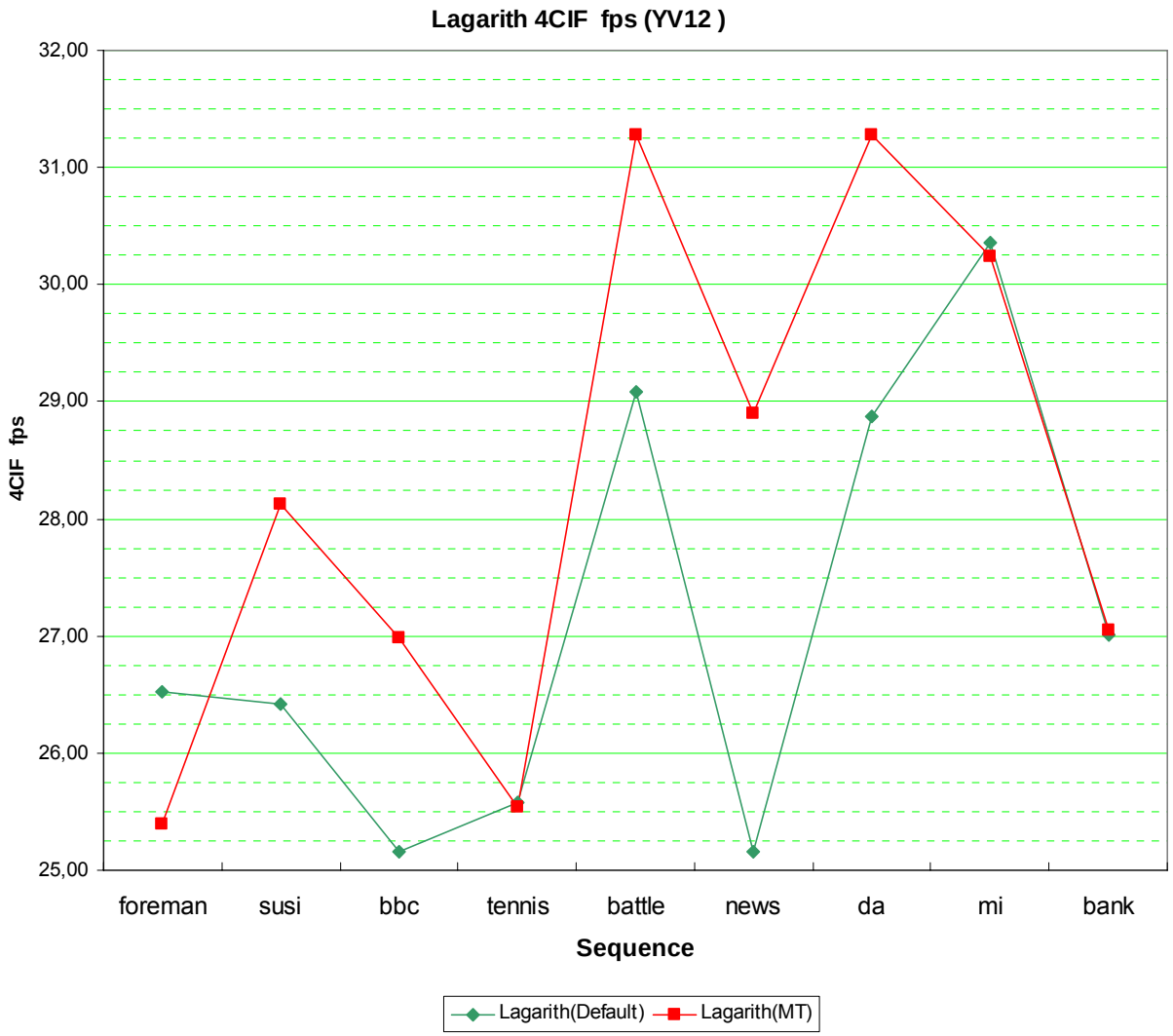


Figure 115. Multithreading: Per-file 4CIF fps for Lagarith, YV12

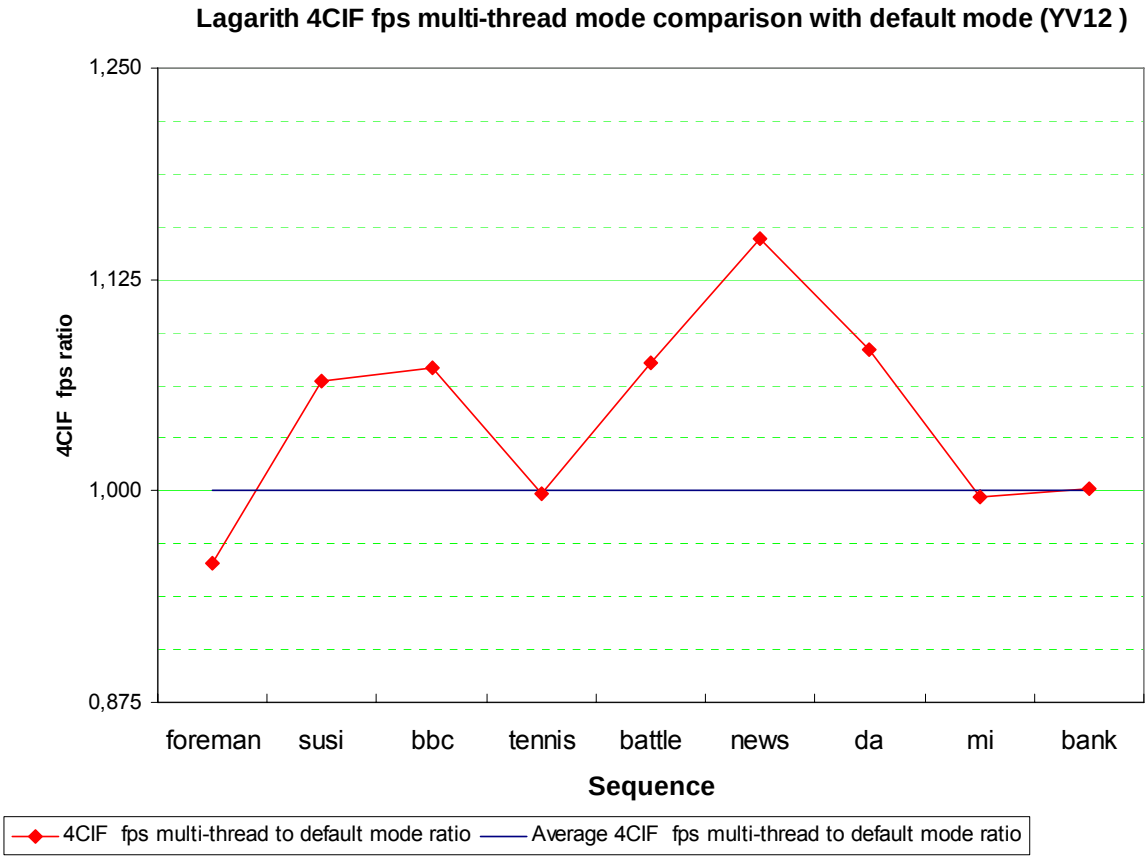


Figure 116. Multithreading: Per-file 4CIF fps for Lagarith in comparison to single-thread mode, YV12

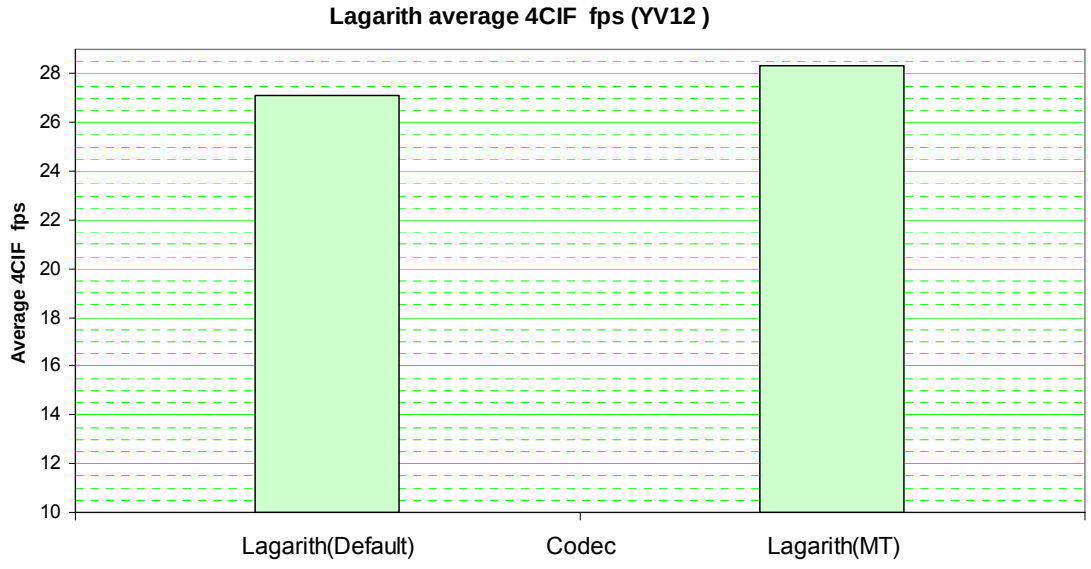


Figure 117. Multithreading: Average 4CIF fps for Lagarith, YV12

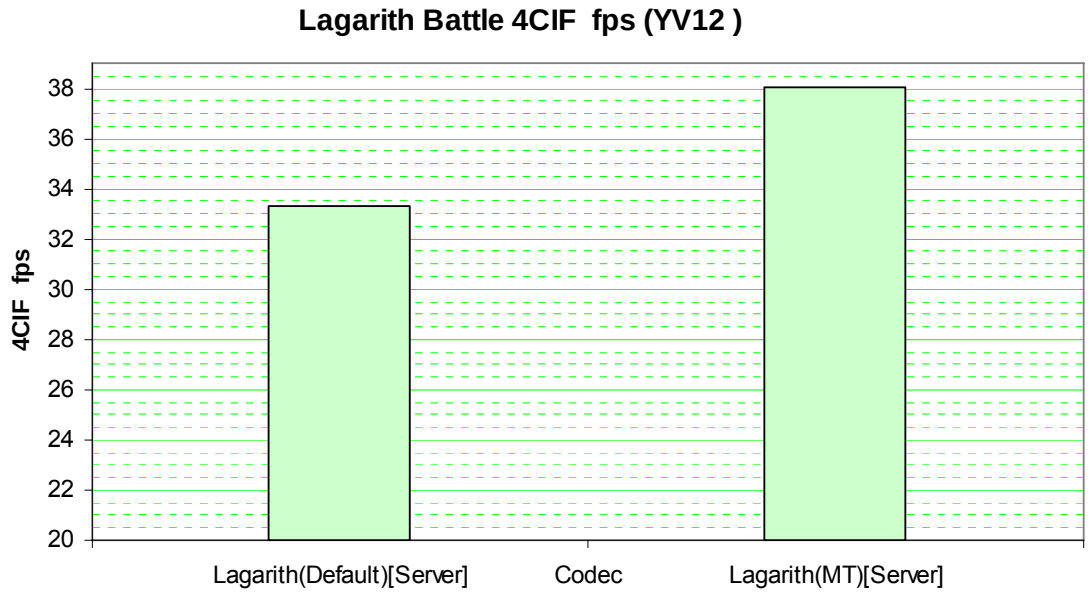


Figure 118. Multithreading: 4CIF fps for Lagarith on "Battle", YV12

4.5.3 x264

Codec	Available relevant parameters	Presets	
		Default	MT
x264		--qp 0 --keyint 300	--qp 0 --keyint 300 --threads 2

Figure 119. Multithreading: Per-file 4CIF fps for x264, YV12

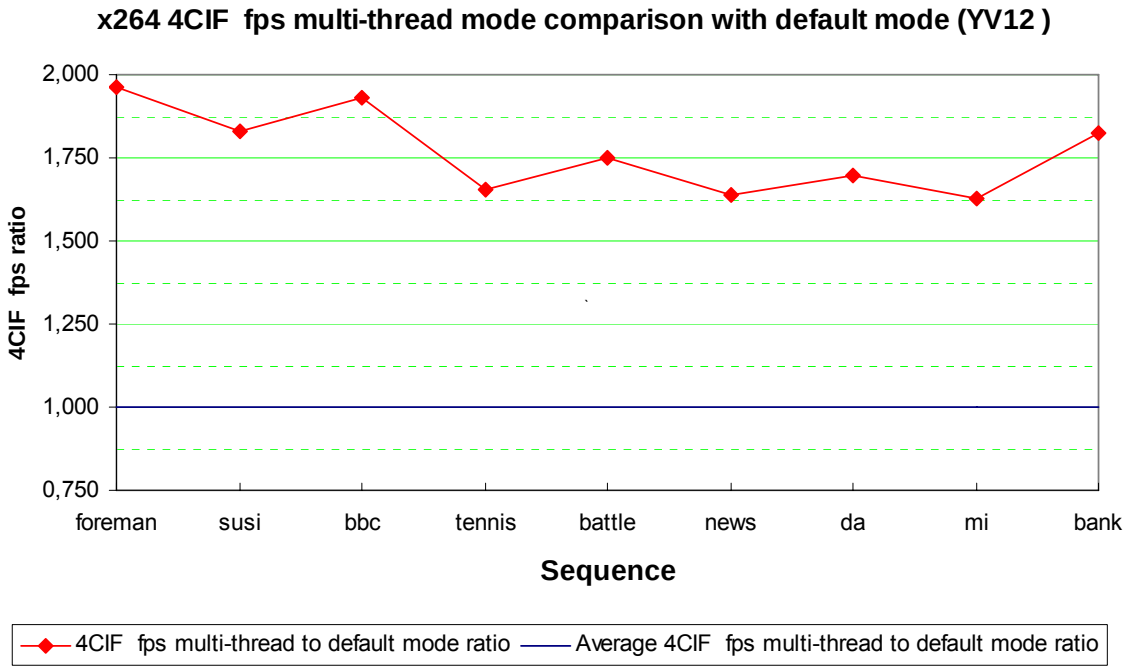


Figure 120. Multithreading: Per-file 4CIF fps for x264 in comparison to single-thread mode, YV12

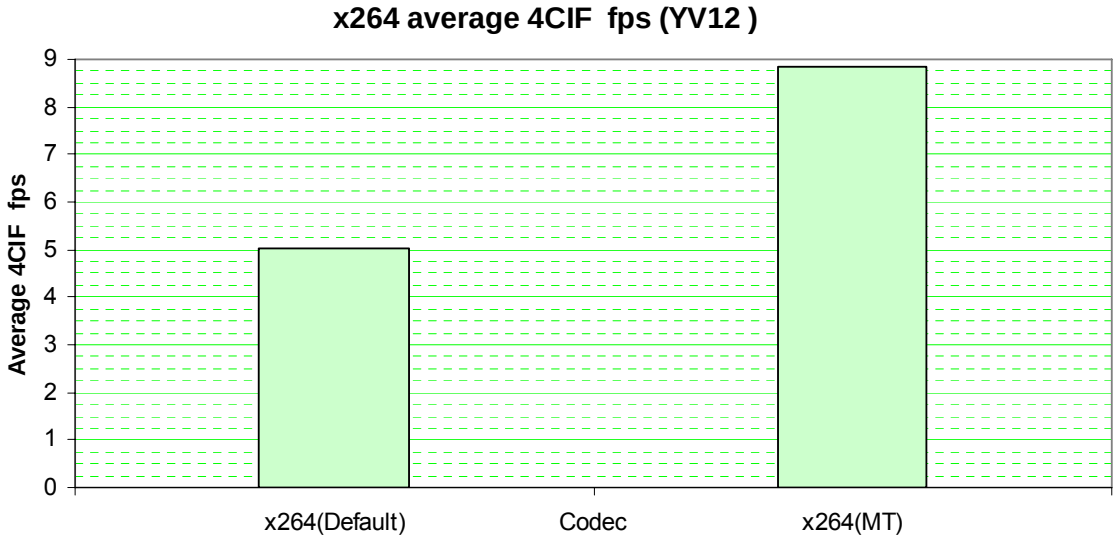


Figure 121. Multithreading: Average 4CIF fps for x264, YV12

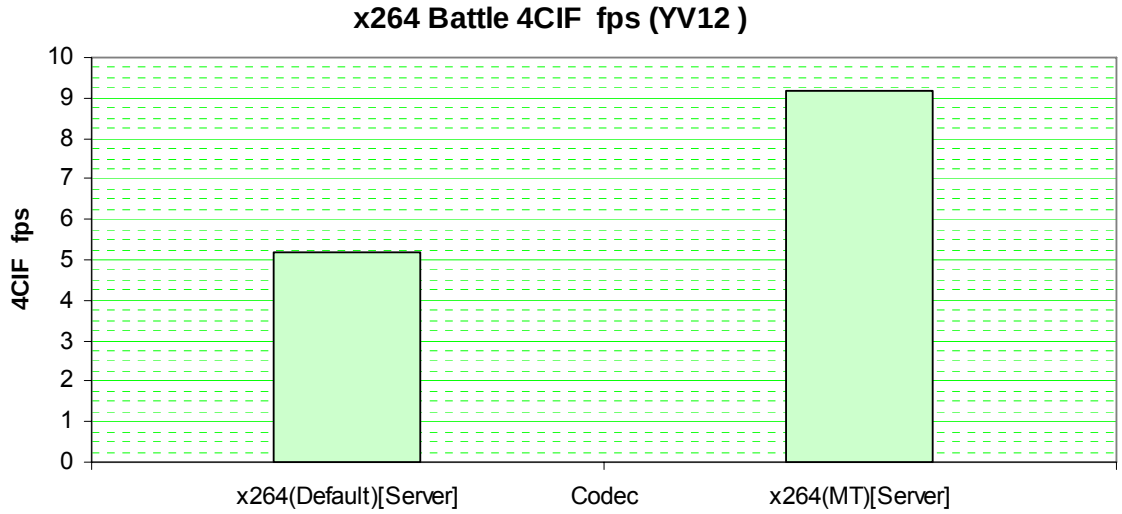


Figure 122. Multithreading: 4CIF fps for x264 on "Battle", YV12

4.5.4 Analysis and Conclusions

The table below summarizes MT performance results.

Codec	Color space	Speed improvement, %
AviZlib	RGB24	9
Lagarith	RGB24	4
Lagarith	YUY2	4
Lagarith	YV12	4
x264	YV12	77

The following conclusions can be made:

- **AviZlib**

This codec shows very strange multi-threading behaviour. In half of cases single-threaded mode is significantly faster (up to 53%) than the multi-threaded. In general multi-threaded mode is slightly better though, but this favorable result may be due to test set selection only.

- **Lagarith**

In almost all cases multi-threaded mode was at least not slower than the single-threaded. However there isn't much gain from multi-threading. Most of time second CPU wasn't used, from time to time it was used, but only by a tiny percent, not using full capacity of the system. It seems there is a lot of space for improvement in multi-threading mode in Lagarith.

- **x264**

This codec fully used capacity of the system. Both CPU were almost fully occupied. The advantage from using multi-threading is stable for all sequences, and in general it speeds up the codec 77%. Still this doesn't

change considerably position of x264 in relation to other codecs by compression/speed trade-off. This position is quite low.

5 Global Conclusions

The following conclusions were drawn from the analysis of results acquired for the chosen test set:

1. **In Video Capture and Video Editing Area the overall clear winner is Lagarith.**
2. **In Maximum Compression area the overall winner is YULS.**
3. **The most balanced and flexible codec is FFV1: relatively good speed and high compression for various presets.**
4. In RGB24 the obvious leader for video capturing and editing is Lagarith, it significantly outperforms competitors in this area both by speed and compression ratio. In YUY2 and YV12 situation in video capturing and editing slightly changes. HuffYuv becomes fastest, however the difference in speed with Lagarith is small, while the last is much better in compression. On some sequences in YUY2 ArithYuv is better in compression than HuffYuv, but in general it is worse. That's strange taking into the account that ArithYuv uses arithmetic coding instead of variable-length codes.
5. In the area of maximum compression the leader is YULS. However, in RGB24 color space FFV1 is almost as good as YULS, while being much faster. But in YUY2 and YV12 the advantage of YULS in compression ratio is significant and obvious. In YV12 MSU also shows a good result filling the gap in speed between slow YULS and FFV1.
6. The example of generally weak x264 results possibly demonstrates the necessity of different approaches to lossy and lossless video compression.
7. At the same time x264 clearly shows that multi-threading techniques can significantly improve speed. However for Lagarith speed improvement is negligible.

The following table summarizes comparison main results showing characteristics of the leaders and theirs closest competitors including HuffYuv as a reference codec for the entire test set (results of leaders are taken for 100%):

Codec	Color space	Compression ratio	Encoding speed, 4CIF fps
Video Capture and Video Editing			
Lagarith "Default"	RGB24	2,79 100%	13,43 100%
	YUY2	2,92 100%	20,80 100%
	YV12	2,55 100%	27,13 100%
HuffYuv "Max"	RGB24	2,15 77%	12,89 96%
	YUY2	2,79 95%	21,37 103%
	YV12	2,49 98%	28,24 104%
Maximum Compression			

Codec	Color space	Compression ratio	Encoding speed, 4CIF fps
YULS “Max”	RGB24	3,31 100%	0,65 100%
	YUY2	3,60 100%	0,76 100%
	YV12	3,14 100%	0,84 100%
FFV1 “Max”	RGB24	3,30 99.8%	6,31 972%
	YUY2	3,41 95%	10,41 1370%
	YV12	2,92 93%	12,41 1479%
MSU “Max”	RGB24	3,11 94%	0,08 13%
	YUY2	3,42 95%	0,07 9%
	YV12	3,03 96%	0,07 9%
HuffYuv “Max”	RGB24	2,15 65%	12,89 1984%
	YUY2	2,79 77%	21,37 2812%
	YV12	2,49 79%	28,24 3365%

6 Acknowledgement

The authors thank Dmitriy Kovalev, Zarxrax and akupenguin for reviewing, valuable comments and suggestions which helped to make this report better.

7 Appendix

7.1 RGB24 compression ratio

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	2,28	1,67	1,46	1,34	2,44	1,90	2,91	3,54	2,14
Alpary(Max)	2,48	2,05	1,65	1,52	2,83	2,17	3,43	4,86	2,40
AviZlib(Fast)	1,44	1,27	1,23	1,19	1,38	1,31	1,75	2,14	1,27
AviZlib(Max)	1,88	1,73	1,47	1,38	2,07	1,97	2,44	3,05	1,69
CamStudio(Fast)	1,62	1,59	1,15	1,43	1,38	1,45	1,80	2,42	1,85
CamStudio(Max)	1,69	1,63	1,15	1,45	1,41	1,54	1,94	2,84	1,92
CorePNG(Default)	1,84	1,73	1,49	1,42	2,10	1,68	2,22	2,53	1,81
CorePNG(Fast)	1,44	1,27	1,23	1,19	1,38	1,30	1,74	2,12	1,27
CorePNG(Max)	2,00	1,91	1,57	1,59	2,39	1,94	2,60	3,25	2,04
FastCodec(Default)	2,15	1,95	1,68	1,57	2,34	1,58	2,64	3,42	2,09
FFV1(Fast)	3,04	2,33	2,02	1,83	3,38	2,49	3,86	5,49	2,86
FFV1(Max)	3,34	2,47	2,13	1,89	3,57	2,78	4,33	6,18	3,03
HuffYuv(Max)	2,37	1,87	1,57	1,31	2,70	1,73	2,53	3,00	2,25
Lagarith(Default)	2,74	2,22	1,85	1,73	3,06	2,24	3,64	5,01	2,65
LOCO(Default)	1,81	1,90	1,60	1,48	2,55	1,96	2,40	2,56	1,98
MSU(Default)	3,15	2,36	2,05	2,01	3,26	2,59	3,40	4,98	2,96
MSU(Fast)	3,03	2,25	1,94	1,92	3,25	2,58	3,56	5,74	3,05
MSU(Max)	3,23	2,38	2,08	2,02	3,36	2,59	3,92	5,42	3,01
PicVideo(Max)	2,16	1,84	1,53	1,38	2,34	1,99	2,62	3,24	1,99
YULS(Fast)	3,55	2,48	2,10	1,96	3,44	2,31	4,40	6,29	3,18
YULS(Max)	3,60	2,48	2,12	1,97	3,44	2,31	4,40	6,29	3,18

7.2 RGB24 4CIF frames per second

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	10,67	12,68	12,46	11,85	14,39	10,97	14,26	13,90	12,05
Alpary(Max)	5,17	5,09	4,43	4,19	5,92	4,55	6,55	7,23	5,34
AviZlib(Fast)	8,62	8,82	9,46	7,88	6,53	4,57	4,73	4,52	4,57
AviZlib(Max)	1,50	4,32	4,17	6,16	1,15	1,39	0,99	0,69	2,36
CamStudio(Fast)	7,67	10,03	7,95	8,08	6,69	8,65	10,48	12,44	11,36
CamStudio(Max)	2,74	5,71	6,75	5,13	4,99	5,56	2,99	0,92	3,39
CorePNG(Default)	4,90	6,39	6,58	6,60	7,68	6,79	7,72	8,21	7,43
CorePNG(Fast)	7,56	8,37	7,13	8,05	9,94	6,69	11,34	9,73	9,95
CorePNG(Max)	0,90	1,76	2,42	1,94	0,67	0,63	0,62	0,41	0,93
FastCodec(Default)	11,05	12,63	12,15	13,25	14,72	12,15	13,93	14,29	12,54
FFV1(Fast)	8,83	8,18	6,63	8,11	8,99	8,14	8,68	9,19	7,55
FFV1(Max)	6,33	4,99	5,01	4,40	7,15	7,15	6,97	9,16	5,67
HuffYuv(Max)	10,40	12,99	11,85	11,52	15,30	13,23	13,76	14,70	12,25
Lagarith(Default)	11,65	13,79	12,71	12,91	13,12	13,17	15,64	15,70	12,15
LOCO(Default)	9,63	9,19	10,25	7,90	12,15	9,52	9,11	11,89	8,89
MSU(Default)	2,49	2,44	2,17	2,22	1,75	1,50	2,54	2,79	2,44
MSU(Fast)	5,38	4,73	4,23	4,34	2,97	2,47	4,18	4,92	5,09
MSU(Max)	0,07	0,10	0,07	0,08	0,08	0,11	0,08	0,11	0,07
PicVideo(Max)	10,69	12,68	12,06	12,02	14,53	10,93	13,64	14,02	11,79
YULS(Fast)	1,29	1,01	1,05	0,91	1,22	0,96	1,32	1,26	1,37
YULS(Max)	0,73	0,81	0,37	0,75	0,44	0,61	0,44	0,89	0,80

7.3 YUY2 compression ratio

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	2,23	1,92	1,66	1,48	2,78	1,89	2,87	3,25	2,34
Alpary(Max)	2,49	2,31	1,92	1,69	3,30	2,19	3,54	4,14	2,68
ArithYuv(Default)	2,54	2,42	2,00	1,82	3,35	2,35	3,46	3,87	2,68
CorePNG(Default)	2,19	2,04	1,79	1,66	2,55	1,93	2,80	3,32	2,23
CorePNG(Fast)	1,80	1,73	1,55	1,47	1,96	1,66	2,21	2,83	1,75
CorePNG(Max)	2,66	2,39	2,04	2,03	3,13	2,23	3,47	4,28	2,81
FastCodec(Default)	2,18	2,26	1,88	1,74	2,59	1,81	2,87	3,38	2,21
FFV1(Fast)	3,09	2,70	2,34	2,03	3,95	2,53	4,06	4,97	3,19
FFV1(Max)	3,32	2,76	2,46	2,06	4,32	2,66	4,36	5,29	3,42
HuffYuv(Max)	2,72	2,47	2,11	1,90	3,39	2,31	3,54	3,82	2,83
Lagarith(Default)	2,75	2,56	2,13	1,92	3,43	2,37	3,78	4,47	2,90
LOCO(Default)	2,74	2,52	2,14	1,92	3,44	2,39	3,74	4,42	2,87
MSU(Default)	3,50	2,85	2,46	2,34	3,83	2,54	3,93	5,27	3,47
MSU(Fast)	3,39	2,79	2,38	2,27	3,92	2,41	3,81	5,19	3,62
MSU(Max)	3,57	2,86	2,50	2,35	4,00	2,56	4,15	5,30	3,49
YULS(Fast)	3,82	2,89	2,45	2,34	4,06	2,79	4,34	5,83	3,69
YULS(Max)	3,91	2,90	2,54	2,34	4,06	2,79	4,35	5,83	3,69

7.4 YUY2 4CIF frames per second

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	18,68	18,63	18,94	18,79	23,99	18,21	22,78	25,17	20,45
Alpary(Max)	7,55	7,96	7,06	6,58	9,63	6,09	9,74	10,38	8,65
ArithYuv(Default)	17,61	20,37	19,47	18,98	23,44	19,29	23,05	25,65	21,50
CorePNG(Default)	11,54	10,93	11,00	9,96	12,59	10,49	13,36	13,84	12,46
CorePNG(Fast)	12,28	16,61	10,48	15,27	12,21	13,97	13,90	23,43	11,88
CorePNG(Max)	0,68	1,06	1,31	1,35	0,54	1,42	0,69	0,80	0,62
FastCodec(Default)	17,73	20,51	20,20	18,89	22,02	18,11	20,57	23,90	19,67
FFV1(Fast)	11,29	13,12	13,31	12,49	13,22	14,88	13,64	16,31	12,30
FFV1(Max)	10,26	8,13	8,21	7,83	13,64	10,88	10,34	15,11	9,31
HuffYuv(Max)	18,97	19,92	19,54	18,84	24,63	21,36	24,71	23,89	20,47
Lagarith(Default)	18,11	20,78	19,56	18,64	22,85	20,98	21,80	23,37	21,09
LOCO(Default)	17,62	14,81	17,13	13,10	20,44	15,85	18,29	20,61	18,61
MSU(Default)	3,61	3,68	3,06	3,12	2,71	3,41	3,66	4,12	3,40
MSU(Fast)	8,19	8,37	7,34	7,15	5,12	6,81	8,94	10,82	8,46
MSU(Max)	0,07	0,07	0,06	0,06	0,07	0,08	0,07	0,06	0,06
YULS(Fast)	1,15	1,38	1,01	1,58	0,91	1,70	0,94	1,52	1,13
YULS(Max)	0,94	1,01	0,40	0,87	0,45	0,87	0,46	0,91	0,93

7.5 YV12 compression ratio

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	1,72	1,85	1,54	1,41	2,51	1,94	2,42	2,72	2,08
Alpary(Max)	1,85	2,22	1,78	1,60	2,91	2,24	2,81	3,17	2,35
CorePNG(Default)	1,77	2,00	1,70	1,60	2,33	1,96	2,32	2,58	2,03
CorePNG(Fast)	1,49	1,63	1,45	1,40	1,81	1,64	1,91	2,18	1,57
CorePNG(Max)	2,10	2,37	1,93	1,96	2,79	2,26	2,83	3,29	2,51
FFV1(Fast)	2,31	2,63	2,19	1,94	3,50	2,61	3,25	3,81	2,81
FFV1(Max)	2,42	2,68	2,31	1,96	3,77	2,73	3,41	4,09	2,96
HuffYuv(Max)	2,09	2,41	1,96	1,81	3,02	2,36	2,98	3,26	2,50

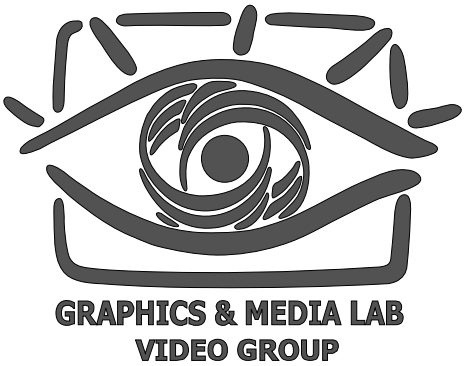
http://www.compression.ru/video/

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Lagarith(Default)	2,11	2,46	1,97	1,84	3,07	2,41	3,07	3,42	2,58
LOCO(Default)	2,12	2,45	1,99	1,84	3,12	2,46	3,07	3,47	2,56
LZO(Default)	1,11	1,18	1,02	0,96	1,51	1,14	1,45	1,90	1,46
LZO(Max)	1,67	1,87	1,53	1,44	2,40	1,87	2,35	2,83	2,17
MSU(Default)	2,69	2,81	2,33	2,25	3,35	2,66	3,29	4,29	3,13
MSU(Fast)	2,61	2,75	2,24	2,18	3,47	2,57	3,20	4,22	3,23
MSU(Max)	2,74	2,81	2,38	2,26	3,55	2,68	3,38	4,31	3,14
Snow(Default)	2,79	2,69	2,36	2,16	3,38	2,65	3,37	3,95	3,02
Snow(Fast)	2,73	2,63	2,32	2,14	3,24	2,61	3,16	3,87	2,98
Snow(Max)	2,83	2,69	2,36	2,16	3,26	2,64	3,25	3,97	3,04
x264(Default)	2,71	2,71	2,30	2,15	2,70	2,19	2,94	4,07	3,04
x264(Fast)	2,29	2,29	1,99	1,89	2,14	1,92	2,52	3,42	2,55
x264(Max)	2,82	2,76	2,33	2,19	2,75	2,28	2,99	4,19	3,10
YULS(Fast)	2,88	2,85	2,31	2,25	3,58	2,88	3,52	4,59	3,28
YULS(Max)	2,95	2,85	2,40	2,25	3,57	2,88	3,53	4,59	3,29

7.6 YV12 4CIF frames per second

codec / sequence	foreman	susi	bbc	tennis	battle	news	da	mi	bank
Alpary(Fast)	23,85	25,96	24,70	25,13	29,41	29,10	29,61	29,53	25,57
Alpary(Max)	6,69	10,46	9,22	8,30	11,85	9,12	11,29	12,24	10,36
CorePNG(Default)	9,82	15,26	14,40	13,13	15,43	12,64	15,81	16,66	15,98
CorePNG(Fast)	21,15	22,10	13,72	20,16	16,33	19,61	16,15	22,41	15,48
CorePNG(Max)	2,04	1,22	1,90	1,72	0,87	0,82	0,99	0,81	0,98
FFV1(Fast)	13,15	17,12	16,31	14,60	16,65	17,95	16,46	17,11	15,70
FFV1(Max)	14,33	10,70	10,84	9,11	14,89	9,16	11,99	17,70	13,00
HuffYuv(Max)	27,12	28,81	26,56	25,71	30,49	27,15	30,19	30,34	27,76
Lagarith(Default)	26,52	26,41	25,16	25,58	29,08	25,16	28,87	30,36	27,01
LOCO(Default)	14,03	17,28	21,44	15,96	19,61	19,47	17,28	23,96	17,98
LZO(Default)	19,67	19,78	21,04	18,27	24,98	18,18	20,84	26,37	23,38
LZO(Max)	6,67	9,34	10,85	8,53	14,74	7,60	13,52	13,91	14,21
MSU(Default)	2,76	4,40	3,46	3,67	3,35	4,24	4,46	4,91	4,05
MSU(Fast)	6,80	10,22	9,21	9,32	6,36	9,32	11,04	13,35	10,80
MSU(Max)	0,07	0,08	0,06	0,08	0,07	0,08	0,06	0,09	0,06
Snow(Default)	1,82	1,95	1,98	1,93	1,54	2,40	2,01	2,40	2,01
Snow(Fast)	6,72	5,09	5,27	4,80	5,19	3,73	5,19	5,30	4,69
Snow(Max)	0,57	0,32	0,34	0,33	0,32	0,30	0,41	0,38	0,37
x264(Default)	4,87	5,12	4,58	4,47	4,95	3,35	5,55	6,53	5,65
x264(Fast)	14,07	10,38	9,09	8,17	11,71	8,17	11,54	13,70	9,72
x264(Max)	1,15	1,21	0,84	1,13	1,01	1,28	1,09	1,39	1,26
YULS(Fast)	1,65	1,33	1,46	1,15	1,31	1,22	1,23	1,16	1,74
YULS(Max)	1,09	1,05	0,44	0,92	0,49	0,96	0,48	1,00	1,11

8 About us (Graphics & Media Lab Video Group)



Graphics&Media Lab Video Group is a part of Graphics&Media Lab of Computer Science Department in Moscow State University. The history of Graphics Group began at the end of 1980's. Graphics&Media Lab was officially founded in 1998. Main research directions of the lab lie in different areas of Computer Graphics, Computer Vision and Media Processing (audio, image and video processing). Some of research results were patented, other results were presented in a number of publications.

Main research directions of Graphics&Media Lab Video Group are video processing (pre-, post- and video analysis filters) and video compression (codecs' testing and tuning, quality metrics research, development of codecs).

Our main achievements in video processing:

- High quality industrial filters for format conversion including high quality deinterlacing, high quality frame rate conversion, new fast practical super resolution, etc.
- Methods for modern TV-sets: big family of up-sampling methods, smart brightness and contrast control, smart sharpening, etc.
- Artifacts' removal methods: family of denoising methods, flicking removal, video stabilization with frame edges restoration, scratches, spots, drop-outs removal, etc.
- Specific methods like: subtitles removal, construction of panorama image from video, video to high quality photo, video watermarking, video segmentation, practical fast video deblur, etc.

Our main achievements in video compression:

- Well-known public comparisons of JPEG, JPEG-2000, MPEG-2 decoders, MPEG-4 and annual H.264 codec's testing; also we provide tests for "weak and strong points of codec X" for companies with bugreports and codec tuning recommendations.
- Our own video quality metrics research, public part is MSU Video Quality Measurement Tool and MSU Perceptual Video Quality Tool.
- We have internal research and contracts on modern video compression and publish our MSU Lossless Video Codec and MSU Screen Capture Video Codec – codecs with ones of the highest compression ratios.

We are really glad to work many years with companies like Intel, Samsung, RealNetworks and others.

A mutual collaboration in areas of video processing and video compression is always interesting to us.

E-mail: video@graphics.cs.msu.ru

http://www.compression.ru/video/

MSU Video Quality Measurement Tool

MSU Graphics & Media Lab. Video Group.

GRAPHICS & MEDIA LAB VIDEO GROUP

Main Features

1. 12 Objective Metric + 5 Plugins

PSNR several versions,
MSAD,
Delta,
MSE,
SSIM Fast,
SSIM Precise,
VQM,
MSU Blurring Metric,
MSU Brightness Flicking Metric,
MSU Brightness Independent PSNR,
MSU Drop Frame Metric,
MSU Noise Estimation Metric,
MSU Scene Change Detector,
MSU Blocking Metric.

2. More Than 30 Supported Formats, Extended Color Depth Support

*.AVI,
*. YUV:
YUV,
YV12,
IYUV,
UYVY,
Y,
YUY2,
*.BMP,
*.AVS:
*.MOV,
*.VOB,
*.WMV,
*.MP4,
*.MPG,
*.MKV,
*.FLV,
etc.,

Extended Color Depth:
P010, P014,
P016, P210,
P214, P216,
P410, P414,
P416,
P410_RGB,
P414_RGB,
P416_RGB.

3. Multi-core Processors Support

MMX, SSE and OpenMP Optimizations

4. Comparative Analysis

Comparison of 3 files at a time

5. ROI Support

Metric calculation for ROI (Region of Interest)

6. GUI & Batch Processing

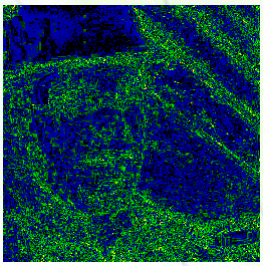
GUI and command line tools

7. Plugins Interface

You can easily develop your own metric

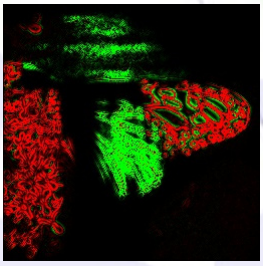
Visualization Examples

Allows easily detect where codec/filter fails

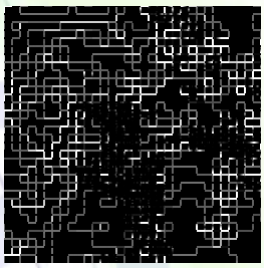


Y-YUV PSNR

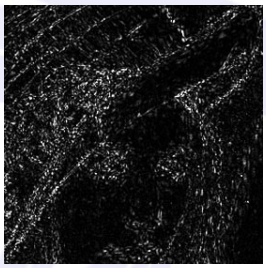
Y-YUV Delta



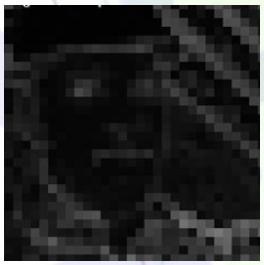
MSU Blurring Metric



MSU Blocking Metric



Y-YUV MSE



VQM

8. Universal Format of Results

Results are saved in *.csv files

9. HDTV Support

10. Open-Source Plugins Available

11. Metric Visualization

Fast problem analysis, see examples above.

http://www.compression.ru/video/quality_measure/index_en.html

Tool was downloaded more than 100 000 times!

Free and Professional versions are available

Big thanks to our contributors:

Apple Inc.
intel
skype
NVIDIA.
NASA
DOLBY
CISCO
Sun microsystems
NBC UNIVERSAL
DivX®