x264 Codec Strong and Weak Points

(Preliminary Options Analysis)

Project head: Dr. Dmitriy Vatolin
Measurements, analysis: Alexander Parshin,
Vladimir Popov,
Kira Ragulina

December 2008

CS MSU Graphics&Media Lab

Video Group

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

x264@graphics.cs.msu.ru

Contents

Introduction

The main purpose of this report is to analyze quality of codecs features implementation using objective methodology. Open source MPEG-4/H.264 codec was chosen as target codec for such analysis because of very good quality of this codec and great number of available options. Specifically we have used *r938* version of x264 codec for our analysis.

Objective quality metrics are used to estimate quality of video quality degradation on single sequence. It is important for us to use automatic metrics calculation because of possibility of massive codec launches.

Preset Analysis Method

The first step of used in this report method is to evaluate objective quality and speed of single codec preset (fixed values for all tested options). Reference preset is used to get relative marks. It is useful when comparing different types of content and codecs. Default preset of x264 codec (no additional presets) is used as reference preset in out report. It means, that both quality and speed of default x264 preset will be equal to 1.0 and all others presets results will be scaled according to this preset results. Estimation of relative quality and relative speed are described below.

Relative Quality Estimation

Quality comparison of single bitrate (compression ratio) is not used because of two reasons:

- Target bitrate should be selected. Any fixed bitrate leads to limitation of target usage area;
- Quality comparison is not correct method if codec has problems with target bitrate keeping.

Instead of single launch quality comparison, we used RD curves comparison.

Given codec's preset and sequence, we can launch codec with several target bitrates and calculate objective quality metrics for each launch. After that we can create approximation of Rate-Distortion (RD) curve (dependence between decoded sequence distortion and encoded stream bitrate). Next, we should compare two RD curves and produce one number as the result of the comparison.

First possible solution is to calculate average metric different between RD curves. This solution is not very good, because of subtracting of metrics values is not always correct. Moreover, sometimes it is difficult to interpret obtained results. For example, is it noticeable difference in results of 0.1 of SSIM quality metric?

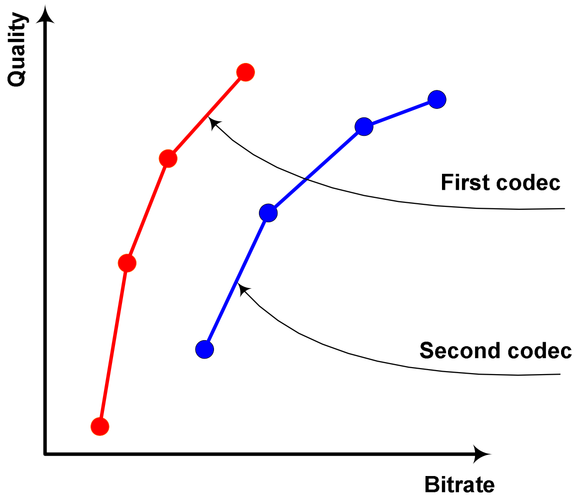
Better method is to work with more correct conception "equal quality". Indeed, if we are interesting in relationship between bitrates for the same quality, we do not need to care about metrics scale and rationality of metric's values subtraction.

We used average bitrates ratio for the same objective quality as main relative mark. There are several stages of its calculation (see Picture 1 – Picture 3):

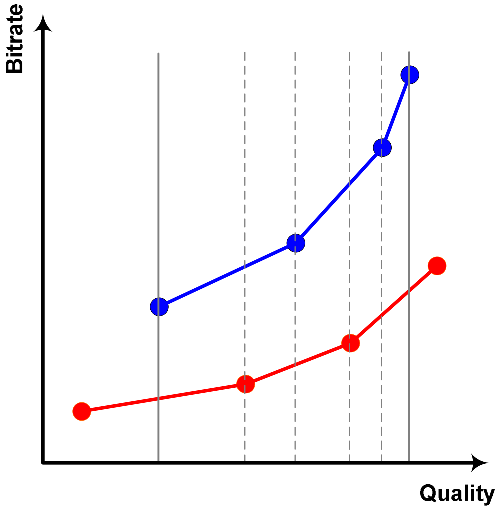
- Initial data is set of RD points for two codecs. We used linear approximation of RD curves.
- First of all, we "rotate" RD curves to simplify future work with bitrate ratio for the same quality. Now we will consider functions *R(D)* instead of *D(R)*.
- Calculating boundaries of averaging. Real RD curves have rather complex form, especially in low bitrates. It is the reason why we don't use extrapolation, working only in areas, where both codecs have estimated RD information. So,

boundaries of calculating are extreme points, where both codecs have RD data (taking into account linear interpolation between real RD points).

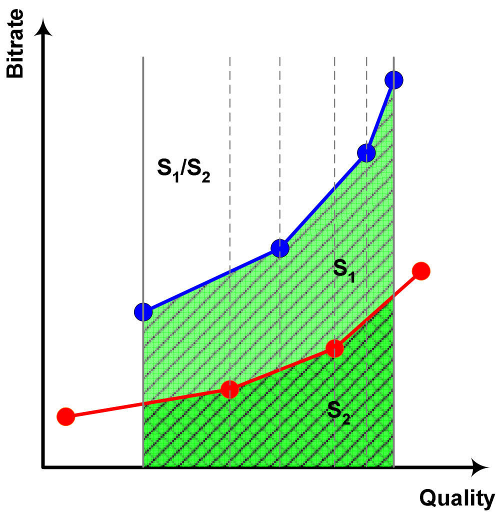
- Bitrate ratio calculation. Ratio of squares below RD curves is used as estimation of average difference between codec results. Linear interpolation between points is used.



Picture 1. Source RD Curves



Picture 2. Axes Rotation and Interval Choosing



Picture 3. Ratio of Squares

Relative Speed Estimation

To get relative encoding time for two presets, we calculate relative time for each sequence and use arithmetic mean to average those values. For each sequences we divide total encoding time of each codec (time to encode sequence with all bitrates) to encoding time of the chosen reference codec. This method allows us to take into account small sequences equally with long sequences (that is the problem of "total encoding time" characteristic).

Glossary

We would use some notions in the report that required some explanations. The following table includes all used terms.

Table 1. List of terms.

	Term	Definition	Example
1.	***Option***	Option is the codec parameter. Codec has a number of options.	number of B-frames, motion estimation algorithm, etc.
2.	***Option value***	Each option has a set of option values. Option value influence on the speed and quality of encoding process.	"—me" option (motion estimation algorithm) has values "dia", "hex", "umh", "esa" and "tesa"
3.	***Preset***	Preset is a set of options with fixed values. If option is missing in presets description, its value is equal to default one.	--me 'dia', --ref 4, --subme 6
4.	***Pareto-optimal point (presets)***	Preset is called pareto-optimal, if there are no other presets that simultaneously give better quality and work faster on given sequences. Number of pareto-optimal presets can be selected for each sequence.	see Picture 6
5.	***Envelope line points (presets)***	Presets lying on the convex hull. It corresponds to the best presets (when the ratio λ between relative encoding time and bitrate is fixed) for all possible ratio λ. See Picture 16.	see Picture 7
6.	***Parameter λ***	Represents desired ratio between relative time and bitrate. Common measure of preset quality can be defined as $M=\lambda T+Q$, where T is relative encoding time and Q is relative encoding quality (see section "Preset Analysis Method" for more details.	

Options and Option Values

We can't test all presets even on the one sequence because the number of possible presets is too large and it is very time-consuming task. So we've chosen only some of them to our analysis.

We have chosen many different options and its values (mentioned in the table below) in order to select optimal presets and analyze options themselves.

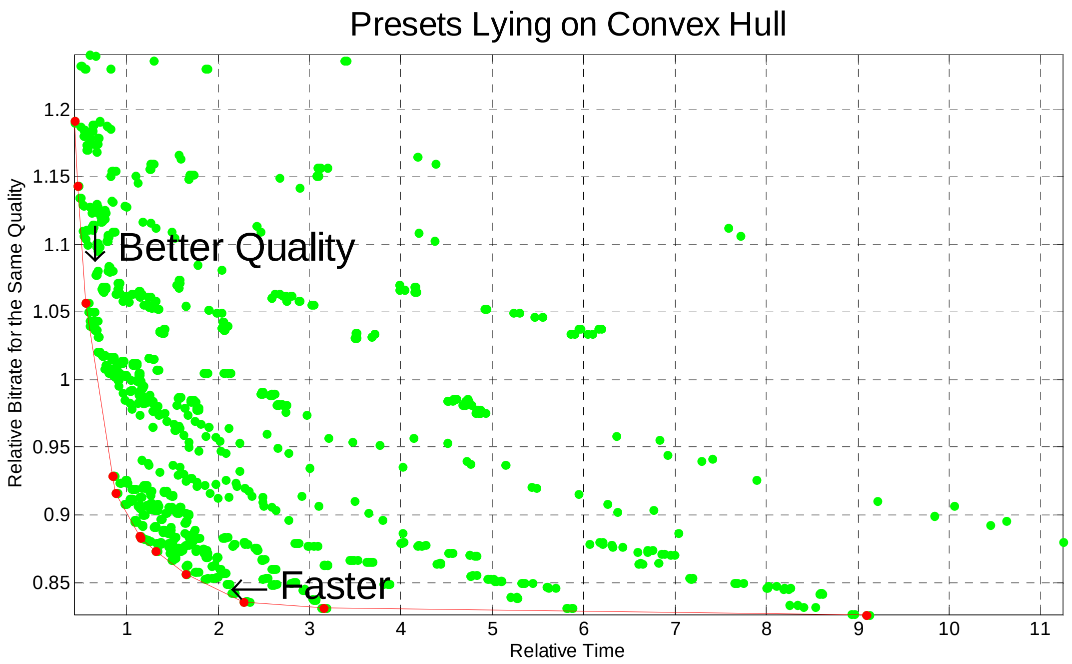
Table 2. List of analyzed x264 options

	Option	Option Values	Comments
1	**Partitions** --partitions x (where 'x' is the partition search types)	"none" "p8x8,b8x8,i8x8,i4x4" "all"	These options determine the partition search types. Default value is "p8x8,b8x8,i8x8,i4x4".
2	**B-Frames** --bframes n (where 'n' is the number of B-frames)	0 2 4	Selects the number of consecutive B-frames between I and P x264 should use. B-frames are frames that are small in size, but when placed correctly, quality loss is insignificant. This can help improve compression effectiveness. Default value is 0.
3	**Reference Frames** --ref n (where 'n' is the number of reference frames)	1 4 8	Selects the maximum number of reference frames that can be used. Reference frames are the frames that refer to other frames (i.e. if both frames are similar) from which they may be predicted. Having a high number of referenced frames will improve quality but slow down encoding. Default value is 1.
4	**Motion Estimation Method** --me x (where 'x' is the motion estimation method)	"dia" "hex" "umh" "tesa"	This option selects the way motion is detected. Motion estimation is a technique to reduce temporal redundancy of a video sequence, and thus it improves compression ratio. It tracks differences between scenes to allocate the various frame types and bitrates. **Diamond (dia)**: Diamond search, radius 1. It has maximum encoding speed. **Hexagon (hex)**: hexagonal search, radius 2. It has worse speed and better quality then the diamond search. **Multi Hex (umh)** (also known as "Uneven Multi-Hexagon"): It is tradeoff between speed and quality. **Hadamard exhaustive(tesa)**: Hadamard exhaustive search. It is slowest method. Default value is "hex".

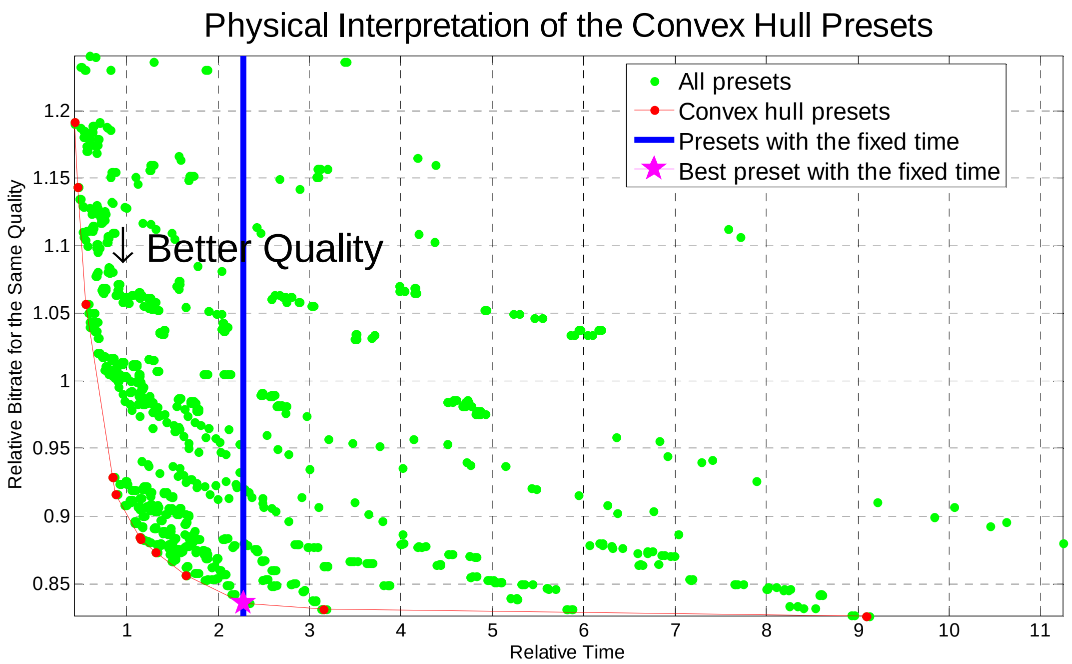
5	**Subpixel Motion Estimation** --subme n (where 'n' is the estimation value)	1 4 5 6	Also known as "Partition Decision". A very important option that determines how x264 makes decisions about motion estimation. The options are available from 1 to 7, with 1 being the fastest (lowest quality) and 7 being the slowest (best quality). Default value is 5.
6	**Mixed References** --mixed-refs (enables mixed references)	off on	This option allows x264 to have greater control over "Reference Frames". Option only available when at least two reference frames has been set. Default value is “off”.
7	**Weighted Prediction** --weightb (enables weighted prediction)	off on	Turns on weighted prediction for B-frames, which results in improved accuracy and therefore in more efficient encoding. Option only available when at least two B-frame has been set. Default value is “off”.

The following Picture 4 shows all tested presets, obtained after enumeration all combinations of mentioned above option values.

Best presets have smaller abscissa (time coordinate) and smaller ordinate (bitrate coordinate). Thus the closer preset to the left lower corner – the better it is. If we fix relative encoding time (bitrate) then optimal for this time (bitrate) presets would be lying on the convex hull (or envelope line). See Picture 5. Convex hull presets correspond to the smaller encoding time (worse quality) are lying more left in the chart and envelope line presets corresponding to the larger time (better quality) are lying more right in the chart. See Picture 4.



Picture 4. All considered presets and axes interpretation.



Picture 5. Physical Interpretation of the Convex Hull Presets.

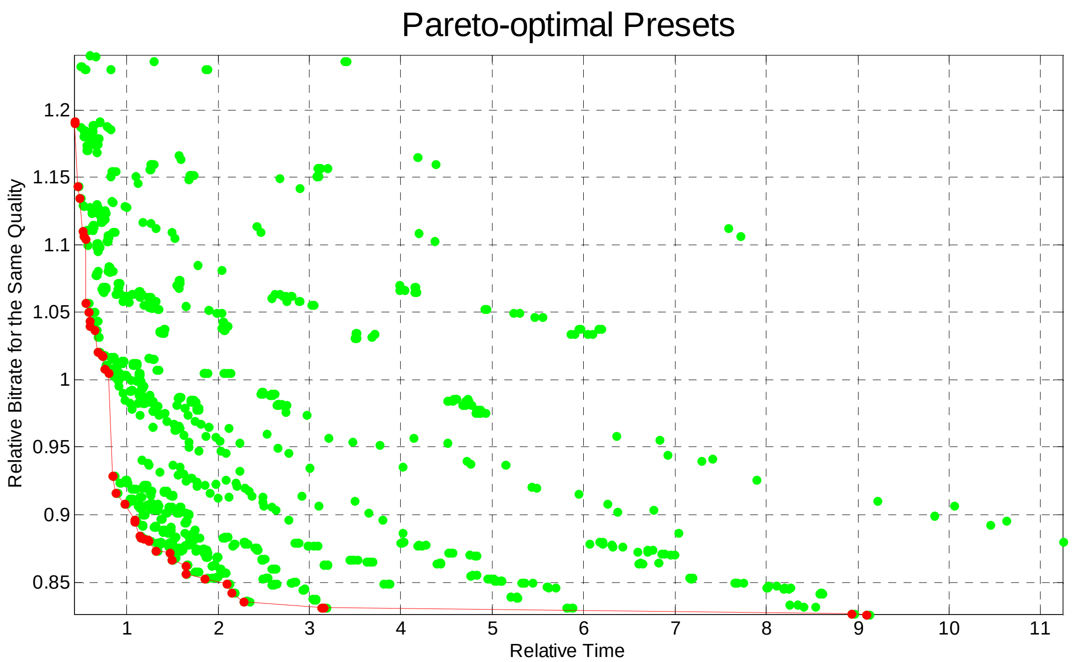
Default preset of x264 codec is used as reference preset in this report. It means that both relative quality and speed of default x264 preset are equal to 1.0 and all others presets results are scaled according to this preset results. The following table demonstrates default preset option values.

Table 3. Default x264 Preset.

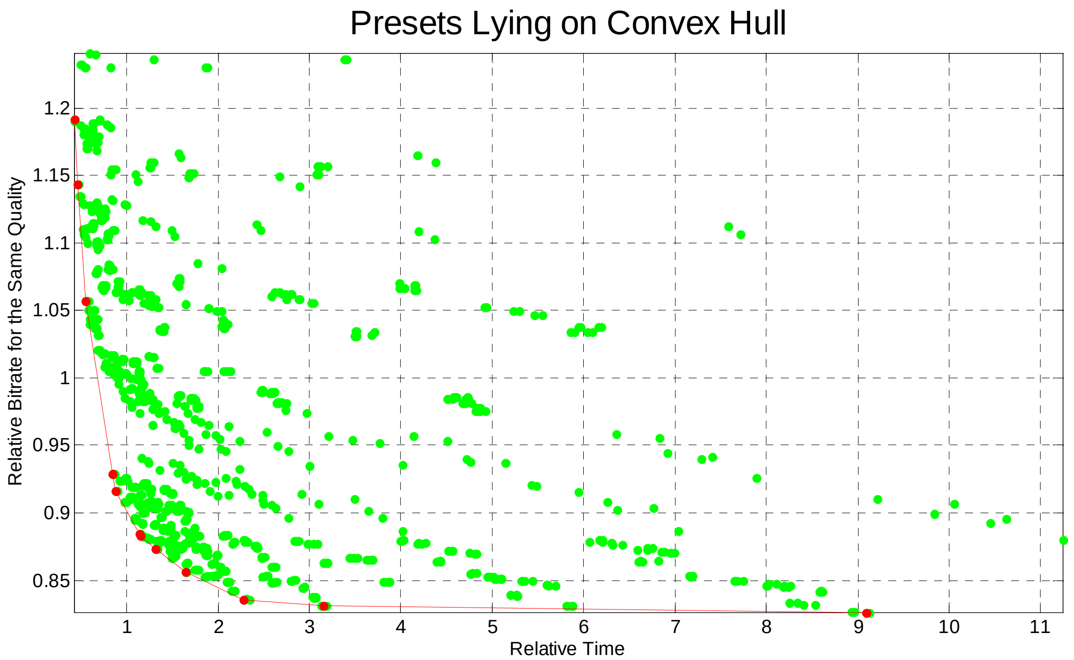
	Option	Option Values of Default Preset
1	**Partitions**	"p8x8,b8x8,i8x8,i4x4"
2	**B-Frames**	0
3	**Reference Frames**	1
4	**Motion Estimation Method**	"hex"
5	**Subpixel Motion Estimation**	5
6	**Mixed References**	off
7	**Weighted Prediction**	off

Best Presets

In this section we show pareto-optimal presets (presets for which there is no other preset, which gives better quality and works faster simultaneously on given sequence) and envelope line presets (i.e. presets lying on convex hull and being the best preset for some ratio λ between relative encoding time and bitrate). Also we will analyze presets lying on the convex hull.



Picture 6. Pareto-optimal presets.



Picture 7. Convex hull presets.

Convex Hull Presets

Table 4. List of Convex Hull Presets.

	Time	Bitrate	--partition	--b-frames	--ref	--me
1.	0.423525	1.191290	"none"	0	1	"dia"
2.	0.460319	1.143350	"none"	2	1	"dia"
3.	0.553577	1.056690	“p8x8,b8x8,i8x8,i4x4”	2	1	"dia"
4.	0.844114	0.928896	“p8x8,b8x8,i8x8,i4x4”	2	1	"dia"
5.	0.882376	0.916337	“p8x8,b8x8,i8x8,i4x4”	2	1	"hex"
6.	1.311870	0.873373	“all”	2	1	"umh"
7.	1.145420	0.884237	“p8x8,b8x8,i8x8,i4x4”	2	4	"hex"
8.	1.154520	0.883289	“p8x8,b8x8,i8x8,i4x4”	2	4	"hex"
9.	1.646830	0.856413	“all”	2	1	"umh"
10.	2.275270	0.835395	“all”	2	4	"umh"
11.	3.154320	0.830835	"all"	2	8	"umh"
12.	9.091010	0.826391	"all"	2	8	"tesa"

Table 5. List of Convex Hull Presets (Continuation).

	Time	Bitrate	--subme	--mixed-refs	--weightb[1]
1.	0.423525	1.191290	1	off	off
2.	0.460319	1.143350	1	off	off
3.	0.553577	1.056690	1	off	off
4.	0.844114	0.928896	4	off	off
5.	0.882376	0.916337	4	off	off
6.	1.311870	0.873373	4	off	off
7.	1.145420	0.884237	4	on	on
8.	1.154520	0.883289	4	on	off
9.	1.646830	0.856413	6	off	off
10.	2.275270	0.835395	6	on	off
11.	3.154320	0.830835	6	on	off
12.	9.091010	0.826391	6	on	off

Summary

- Relative encoding time values variation is greater than in 26 times and relative bitrate values variation is greater than 50% considering from the best value of bitrate.
- Convex hull presets analysis is shown in the following table.

[1] This option is not significant, you can do not choice its value.

Table 6. List of Convex Hull Presets Analysis Results.

Option	A Lot of Presets	A Few Presets	Little Time (High Bitrate)	Middle Time (Middle Bitrate)	Long Time (Low Bitrate)
1. --partitions			"none", "p8x8,b8x8,i8x8,i4x4"	"p8x8,b8x8,i8x8,i4x4", "all"	"all"
2. --bframes	2	0, 4	0, 2	2	2
3. --ref	1		1	1, 4	4, 8
4. --me		"tesa"	"dia", "hex"	"hex", "umh"	"umh","tesa"
5. --subme		5	1, 4	4, 6	6
6. --mixed-refs	off		off	off, on	on
7. --weightb	off	on	off	off, on	off

Colored Clouds Presets Analysis

Method Description

The simplest presets analysis method consists of considering distribution of presets with fixed option value.

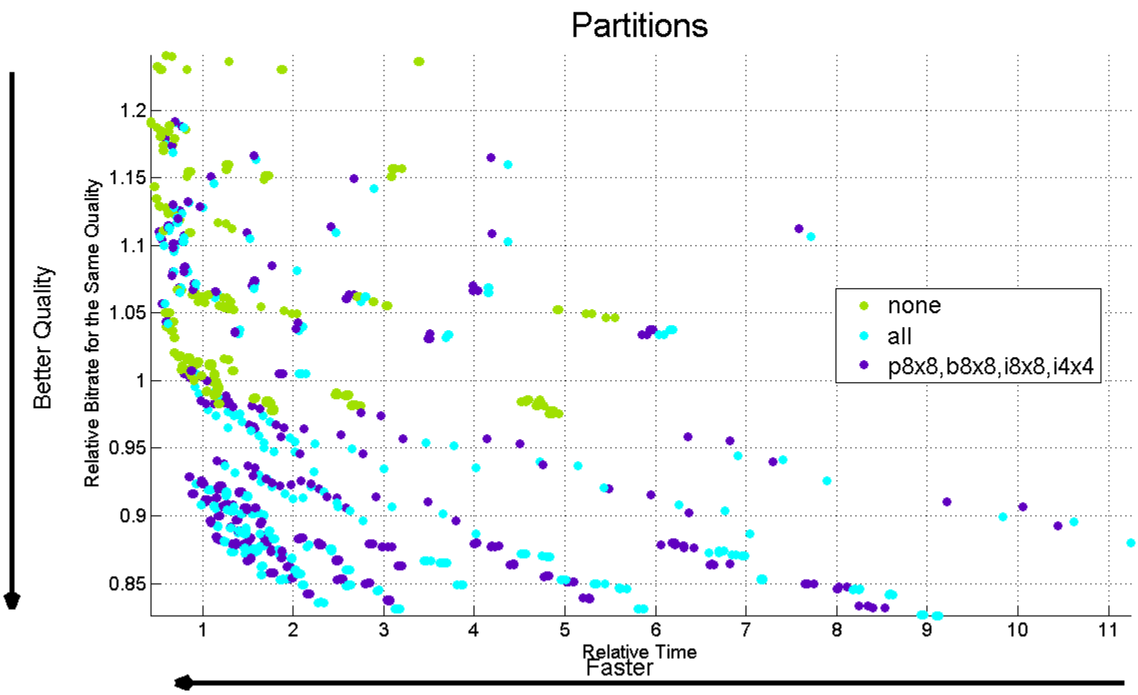
In this section we analyze presets using this method. The following charts have been constructed as follows. We paint all presets with the same value of concerned option in the same color. Thus if two presets have the same value of the considered option they will be paint in the same color and if their values are different then they will be colored in the different colors.

Best presets have smaller abscissa (time coordinate) and smaller ordinate (bitrate coordinate). Thus the closer preset to the left lower corner – the better it is. See Picture 4.

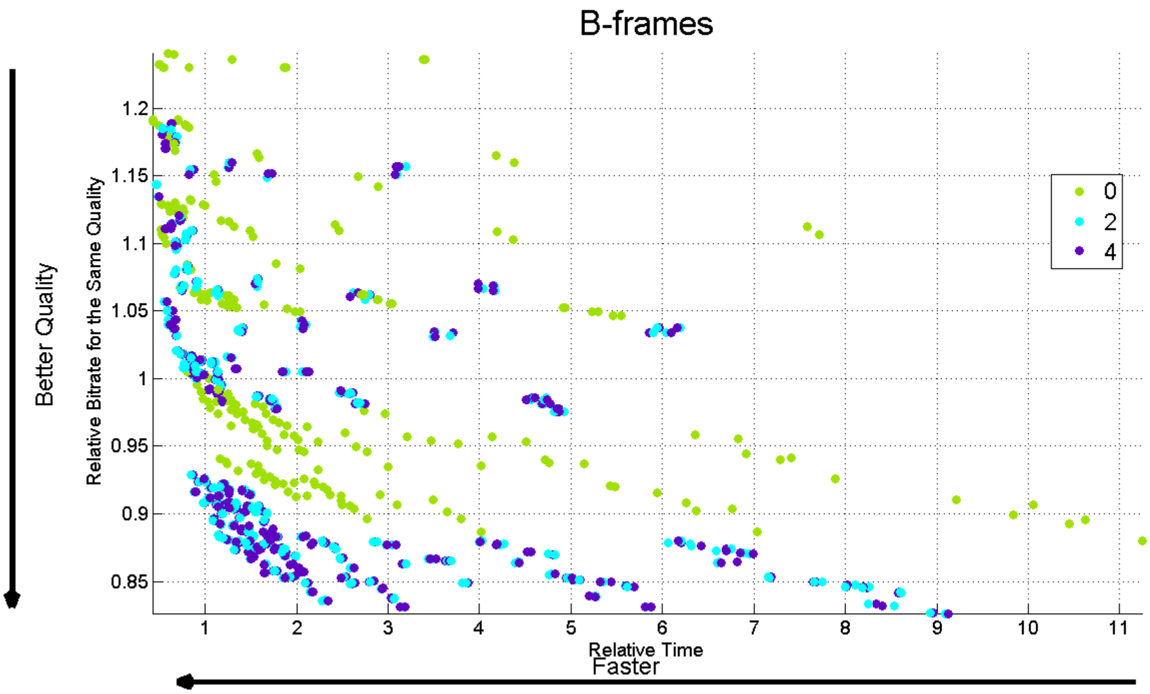
This method has its own highs and lows. Its advantage is clearness. But it is its drawback at the same time, because of subjective perception. That's why we draw colored presets on the chart in the random order to eliminate this drawback.

Charts for all concerned options are shown below (Picture 8 – Picture 14).

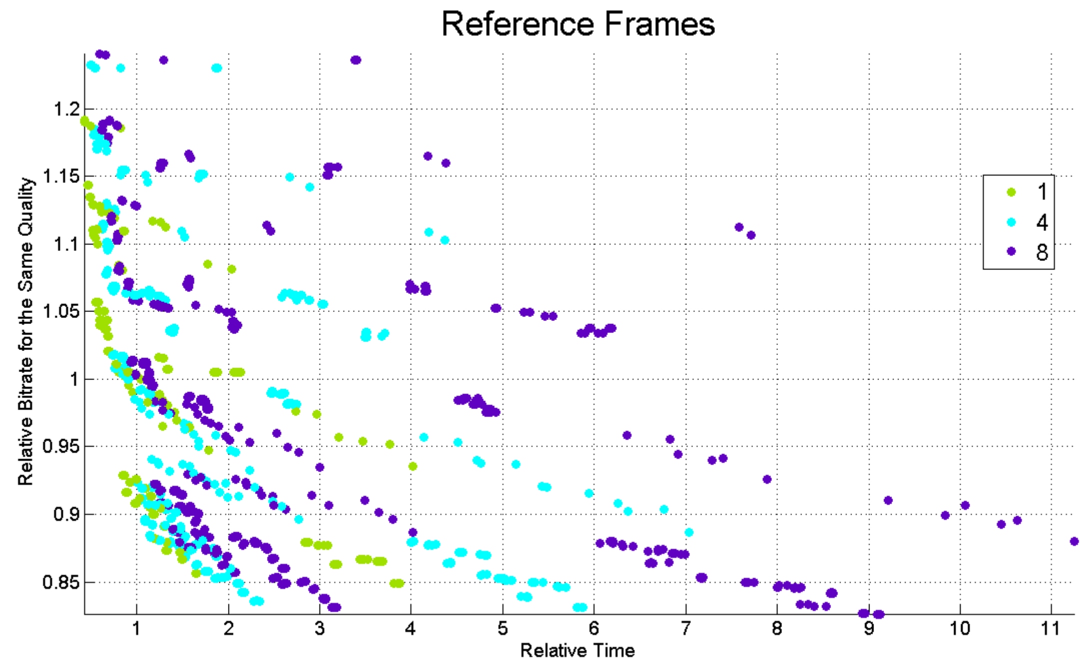
Results



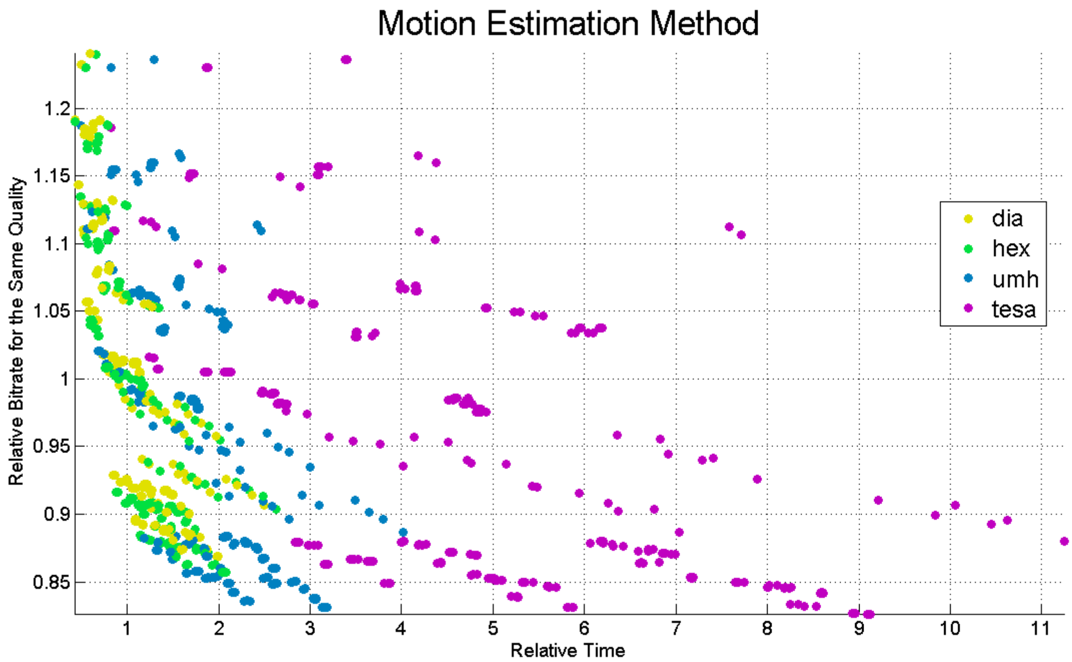
Picture 8. Clouds Presets Analysis of Partitions Option.



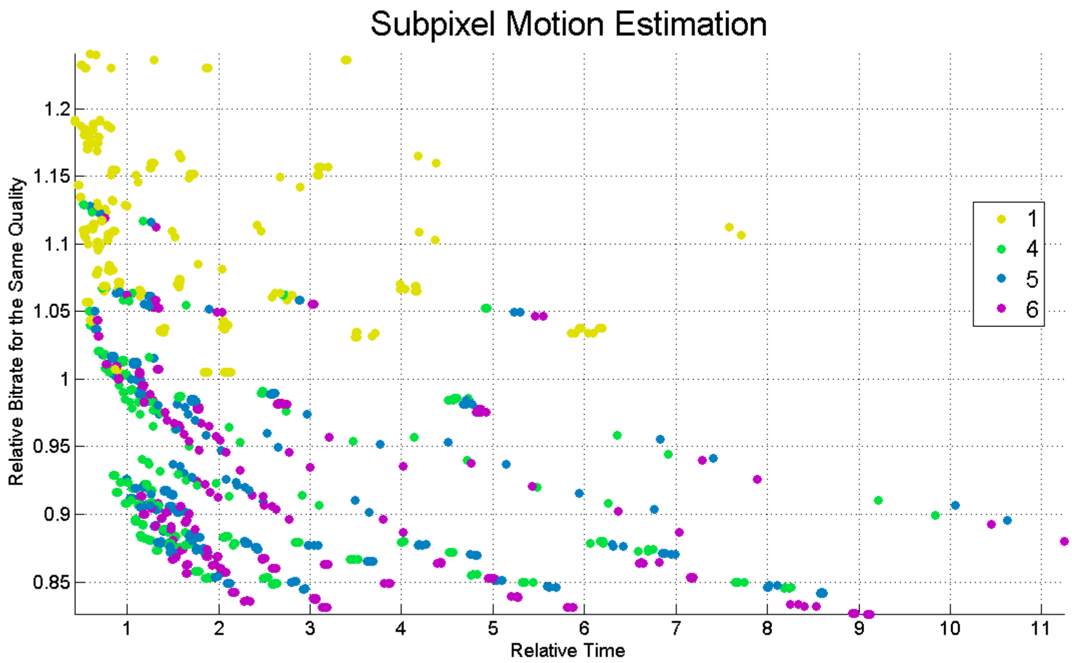
Picture 9. Clouds Presets Analysis of B-frames Option.



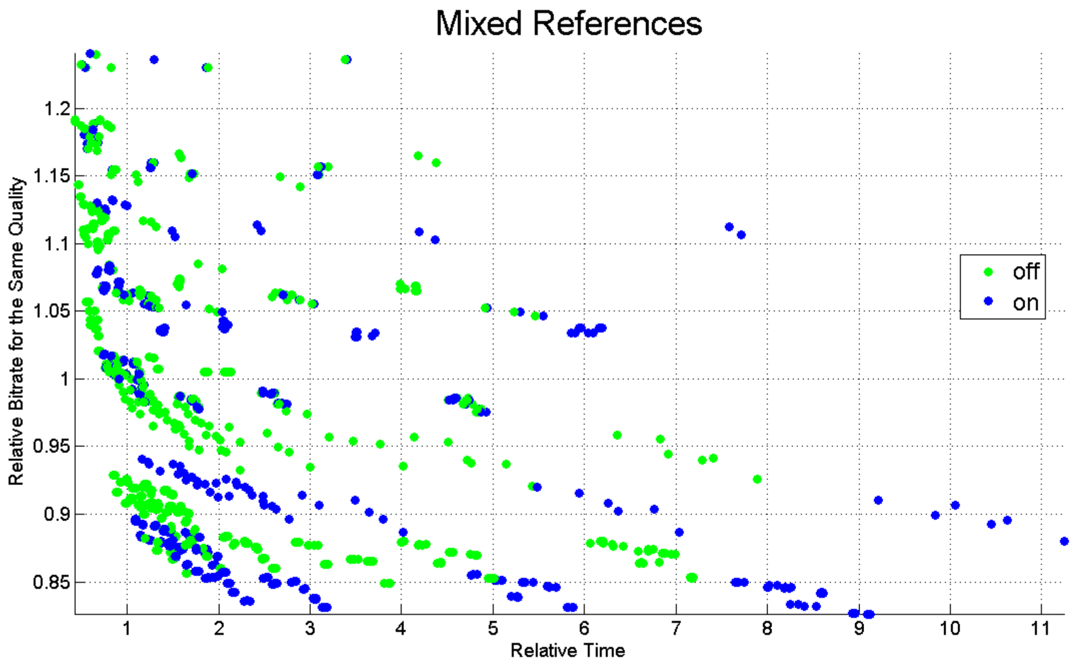
Picture 10. Clouds Presets Analysis of Reference Frames Option.



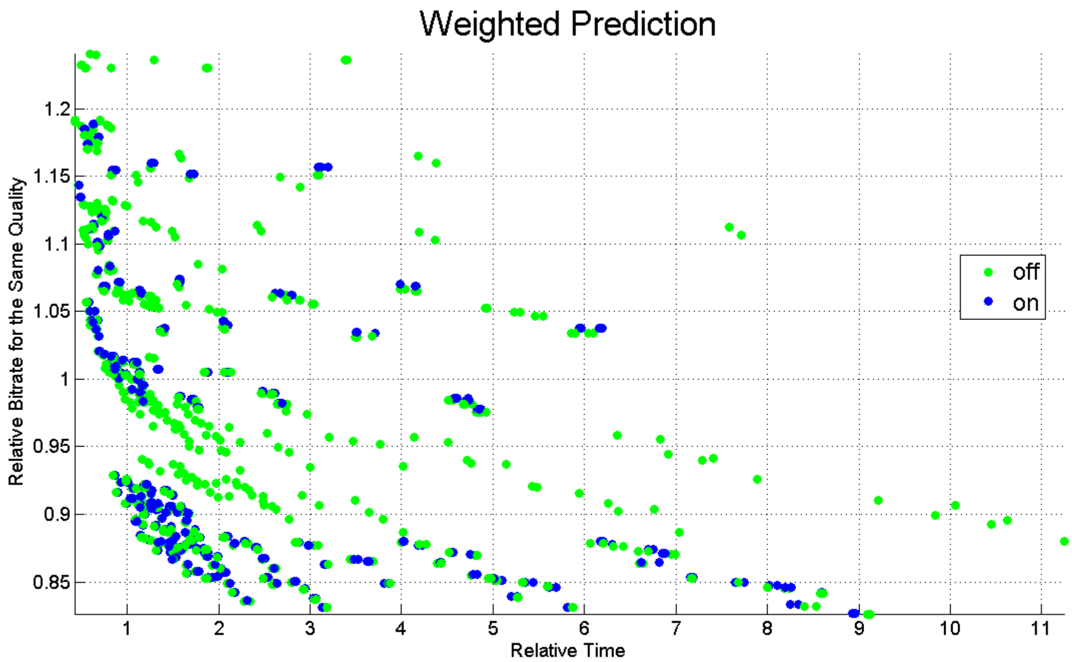
Picture 11. Clouds Presets Analysis of Motion Estimation Method Option.



Picture 12. Clouds Presets Analysis of Subpixel Motion Estimation Option.



Picture 13. Clouds Presets Analysis of Mixed References Option.



Picture 14. Clouds Presets Analysis of Weighted Prediction Option.

Options Analysis

Conclusions from demonstrated above charts are shown in the following table.

Table 7. List of Colored Clouds Presets Analysis Results.

	Option	Preset	Comments
1	**Partitions** --partitions x	• "none"	Presets with partitions equal to "none" works well when encoding speed is high. Partitions value "all" is the best when it is required high quality. If it is important both speed and quality the best choice is "p8x8,b8x8,i8x8,i4x4".
		• "all"	
		• p8x8,b8x8,i8x8,i4x4"	

2	**B-Frames** --bframes n	• 0 • 2 • 4	If maximum encoding speed is required the best choice is 0. In other cases the 2 or 4 B-frame is better. Values 2 and 4 of option B-frames does not significantly differ.
3	**Reference Frames** --ref n	• 1 • 4 • 8	Presets with 1 reference frame is better when the speed is more important than quality. If speed is not the most important factor, but still important 4 reference frames are more preferable. 8 reference frames is optimal when maximum quality is required.
4	**Motion Estimation Method** --me x	• “dia” • “hex” • “umh” • “tesa”	Presets with “dia” and “hex” algorithms are optimal if you want to get high speed. “umh” algorithm is a good tradeoff between speed and quality. “tesa” algorithm is optimal when maximum quality is required.
5	**Subpixel Motion Estimation** --subme n	• 1 • 4 • 5 • 6	Presets with subme 5 are not optimal. Among the best presets with high speed all have subme 1. High quality presets with sumbe 6 have better speed than high quality presets with other subme value. If it is important both speed and quality the best choice is subme 4.
6	**Mixed References** --mixed-refs	• off • on	Presets with turned off mixed references are optimal for high speed encoding. If the maximum quality is required the best choice is to use mixed references.
7	**Weighted Prediction** --weightb	• off • on	Optimal presets have both values of weighted prediction option. There are slightly more presets with weighted prediction “off” among the best presets according to the maximum speed values.

Summary

- Results of analysis of the colored clouds of presets are shown in the table below.
- Weighted prediction options don’t change results significantly.
- Difference of 2 and 4 B-frames usage is not significant.

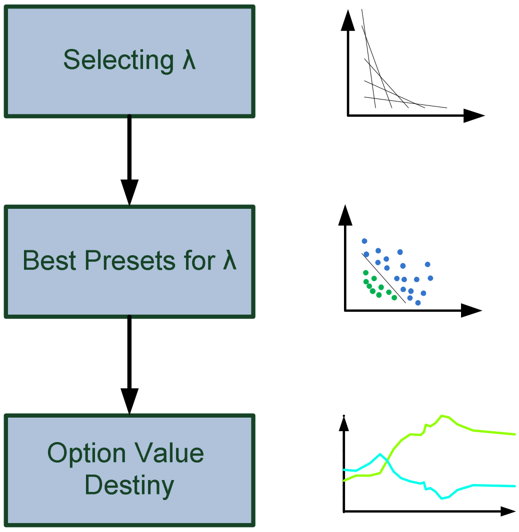
Table 8. List of Colored Clouds Presets Analysis Summary.

	Option	Time Is More Important than Quality	Time/Quality tradeoff	Quality Is More Important than Time
1.	--partitions	“none”	“p8x8,b8x8,i8x8,i4x4”	“all”
2.	--bframes	0	2, 4	2, 4
3.	--ref	1	4	8
4.	--me	“dia”, “hex”	“umh”	“tesa”
5.	--subme	1	4	6
6.	--mixed-refs	off	off, on	on

Lambda Presets Analysis

Method Description

Both speed and quality are important for users when they use some video codec. Unfortunately these two characteristics are very different and it is very difficult to compare them. If one preset has higher speed and quality than another one, we can say that it is better. But what if it has higher speed and worse quality than another preset? It is desirable to have method which allows comparing presets with arbitrary values of speed and quality. Really sometimes we can compare aforementioned presets. Let one preset has twice higher speed and requires additional 0.1% of size for the same quality than another one. It will be logical to accept that first preset is better. To formalize these words we use some accessory parameter *λ*, which represents desired ratio between relative encoding time and bitrate.



In this section we analyze presets using this method and the following algorithm. First of all we define ratio between encoding time and bitare. After that, all presets are ranked using this parameter and 10% of best presets are considered. At the last step we use destiny of presets with option value to analyze current option for different ratios between relative encoding time and bitrate among selected presets. All steps are described below in derails.

Selecting *λ*

This method has its own highs and lows. Its advantage is that it can describe different distribution of presets with different option values in different parts of all measured presets convex hull (ratio between relative encoding time and bitrate). But its drawback is requirement to choice lambdas correctly. It is good idea to choice *λ* such that amount of different presets among best presets for different neighbor *λ* will be the approximately the same. We use this idea and get several values of lambda between 0.01 and 7. We have chosen the number of different best presets among different neighbor lambdas equal to 17 presets. The number of best 10% of presets equals to 120. See Picture 16 – Picture 18 for resulting best presets. On those pictures best presets corresponding to the same value of lambda colored in the same color and different colors correspond to different values of *λ*. Big values of *λ* mean that the speed is more important than the quality (in extreme case, *λ* equal to infinity, quality is not important at all) and low values mean opposite fact.

Best presets selection

For each value of *λ* we consider the quantity $M=\lambda T+Q$ as preset common quality measure. In compliance with this measure we have chosen 10% of the best presets and analyze them (see Picture 15). General question is how many presets with fixed value of some option belong to these 10%? Below in this section we would consider best presets in terms of common quality according to some fixed *λ*.

Thus the more presets with this option value belong to the best 10% of presets - the more preferable this option value for fixed ratio between relative encoding time and bitrate (i.e. *λ*).

Analyzing option value density

The chart for the each option built in the following way. There is one line corresponding to each option value on the chart. For each chosen *λ* value and for each option value we consider the quantity $N(\lambda,k)$ equals to ratio between number of best 10% of presets for this λ with this option value k and total number of presets with this option value k:

$$N(\lambda,k)=\frac{N_k^\lambda}{N_k}, \quad (1)$$

where N_k^λ - number of best 10% presets for this λ with this option value k, N_k - total number of presets with this option value k. Then we divide this quantity $N(\lambda,k)$ by sum of quantities $N(\lambda,k)$ for all possible for this option values m and multiply by 100 to get quantity in percents:

$$N_{\%}(\lambda,k)=100\frac{N(\lambda,k)}{\sum_m N(\lambda,m)} \quad (2)$$

This quantity $N_{\%}(\lambda,k)$ corresponds to the point on the chart with λ and belongs to a line corresponds to the k option value. Such points for all chosen λ values make this line overall.

Some combinations of option values are invalid, for example weighted prediction equals to "on" and b-frames equals to 0. That's why the number of presets with the different values of the same option is various. Therefore we divide the number N_k^λ of best 10% of presets for λ with option value k by total number N_k of presets with option value k. See Formula 1.

According to the definition the sum of all points with the same X Axis is equal to 100%. See Formula 2.

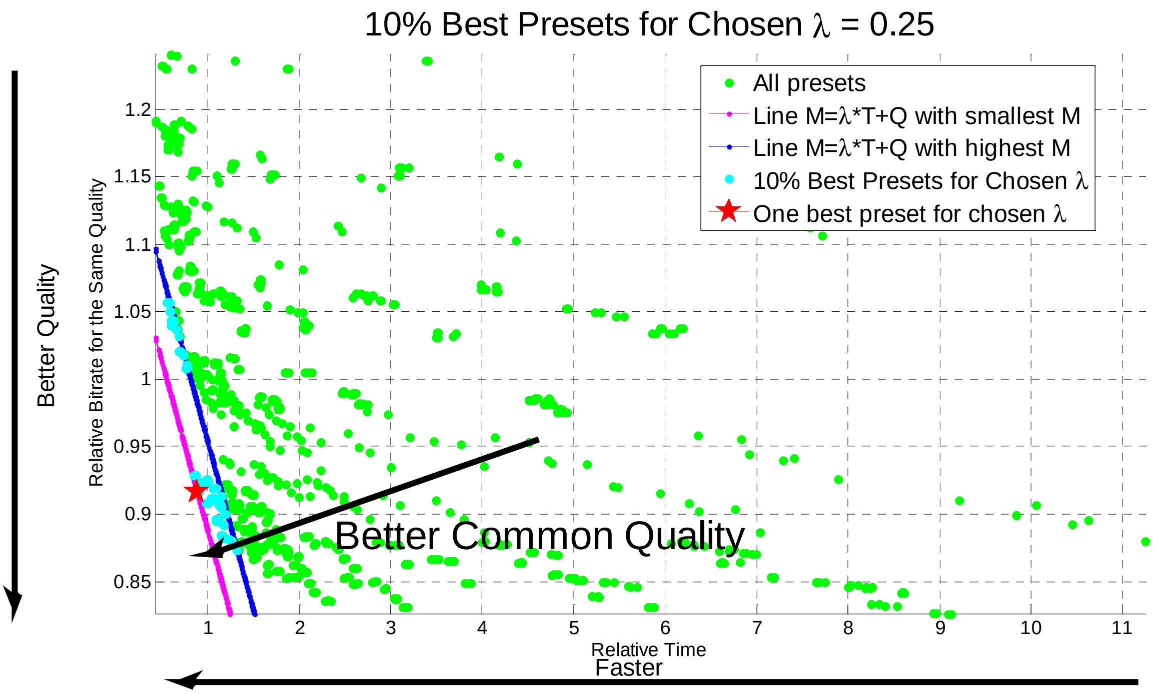
Thus the line corresponding to the option value is higher – the more preferable this option value. For example see Picture 19. If the line has maximum at some λ (value of ratio between relative encoding time and bitrate) it means that it is optimal ratio for option value corresponding to this line.

Below, charts for all concerned options are shown (Picture 19 – Picture 25).

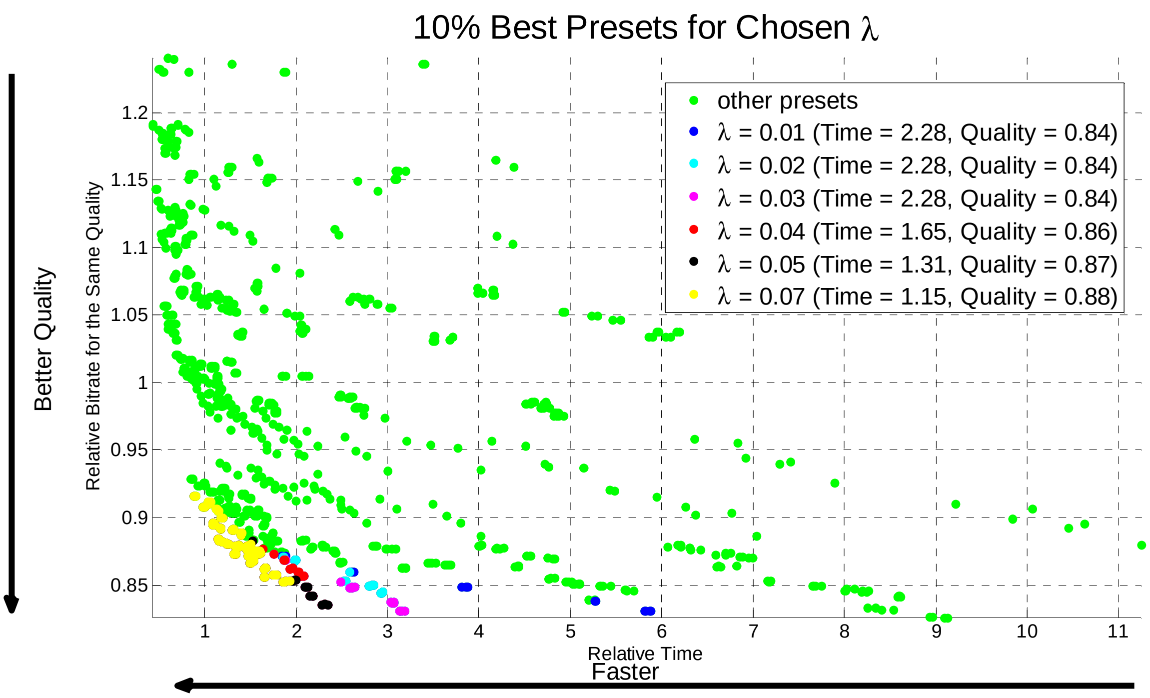
Note that the scale of Y Axis is varying from chart to chart.

Note that X Axis represents the value of lambda (ratio between relative encoding time and bitrate) but not relative time or bitrate in spite of the axis labels. For convenience we use logarithmic scale by lambda (X Axis). There are same value of the time and bitrate corresponding to the different value of lambda. That's why these parameters (time and bitrate) changed discrete in the following charts.

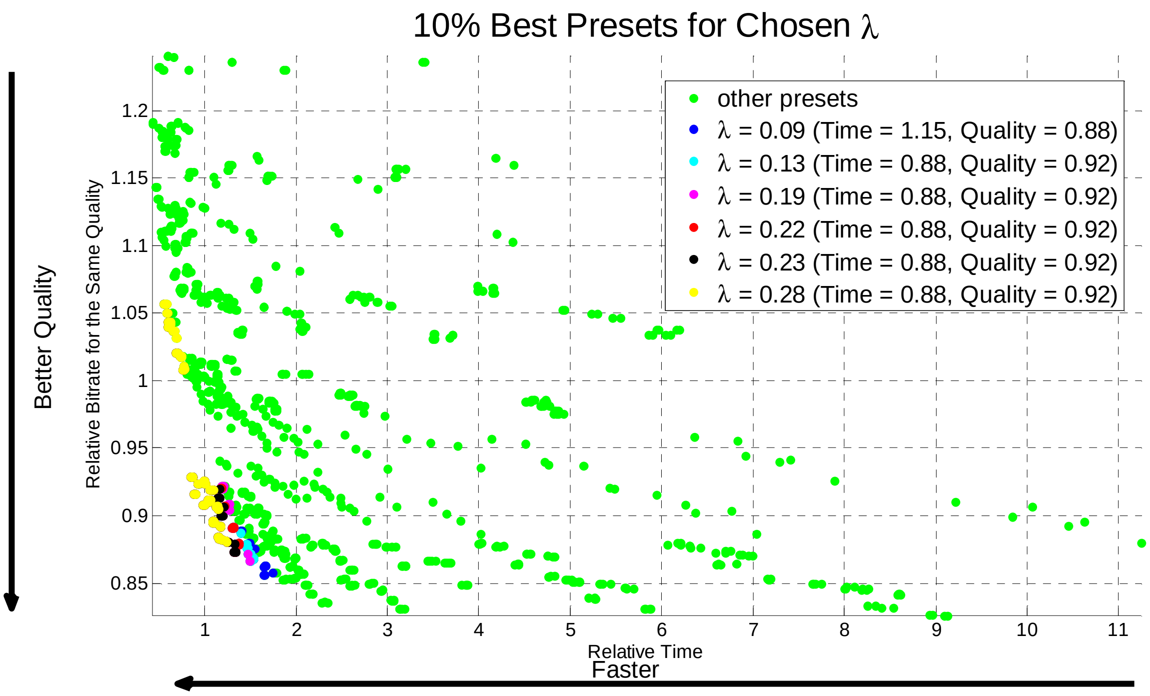
Results



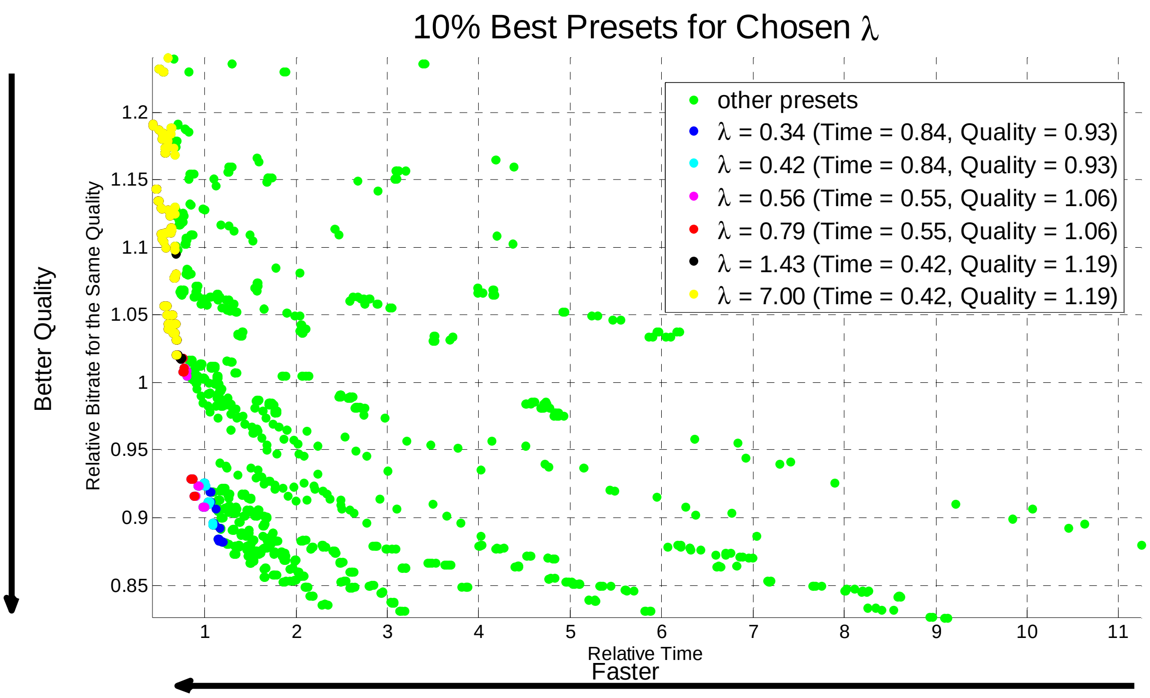
Picture 15. Explanation of measure based on *λ*.



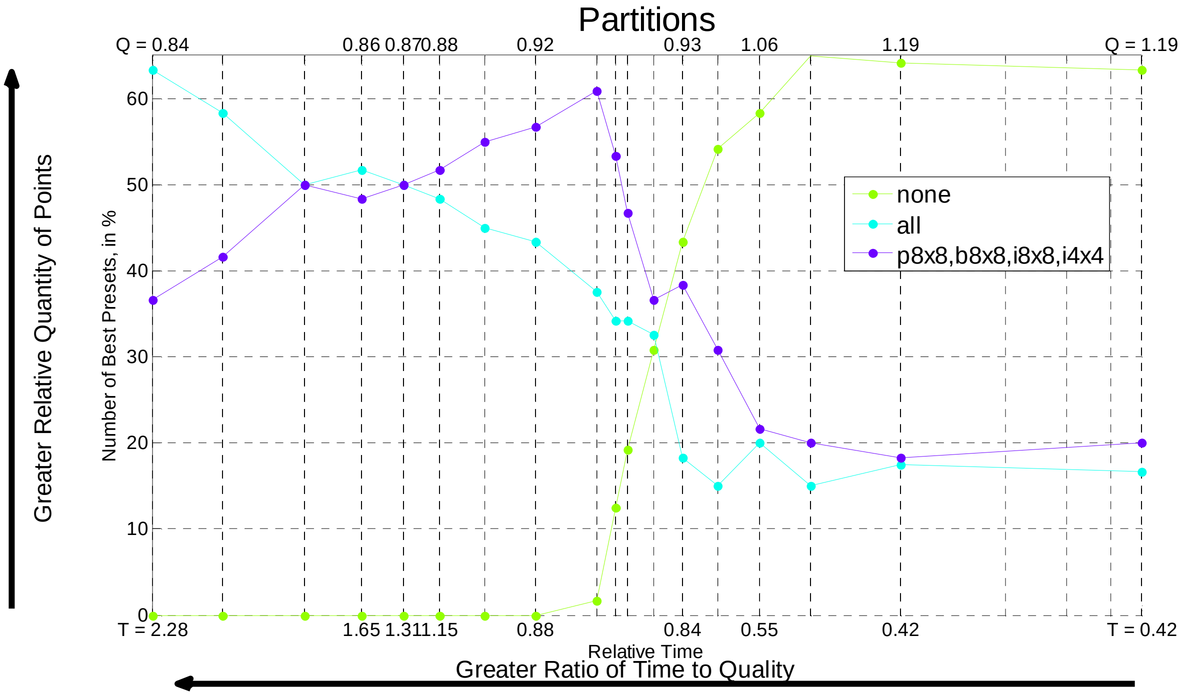
Picture 16. Distribution presets on classes in *λ* presets analysis.



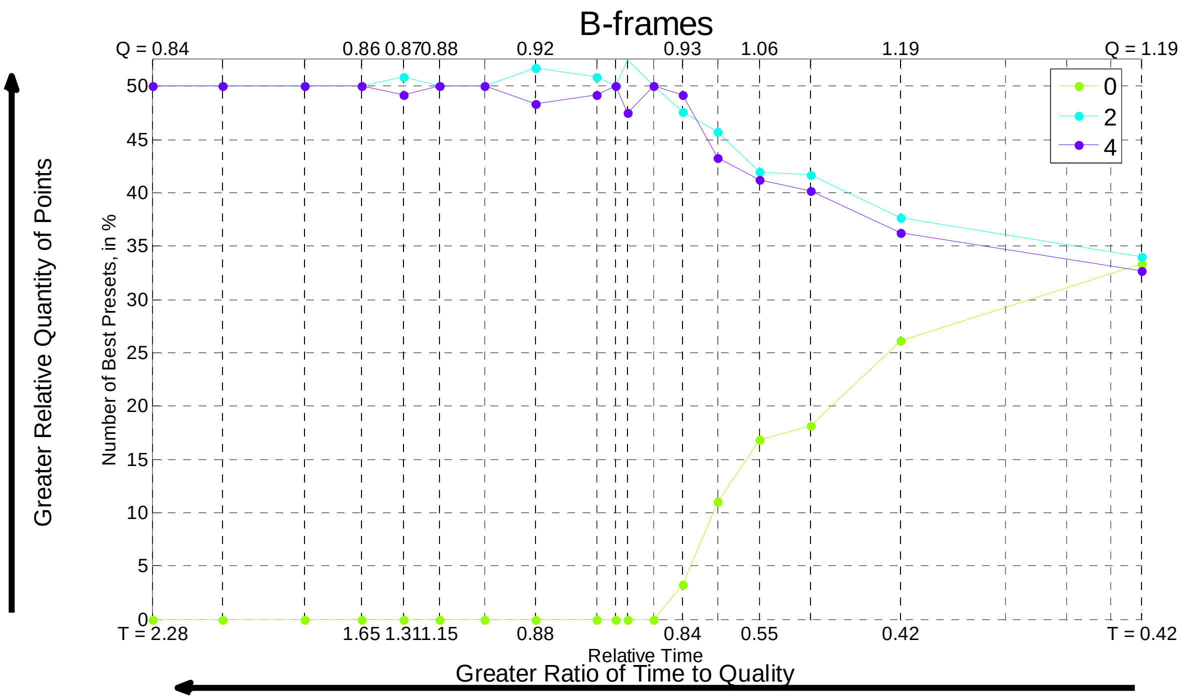
Picture 17. Distribution presets on classes in λ presets analysis (Continuation).



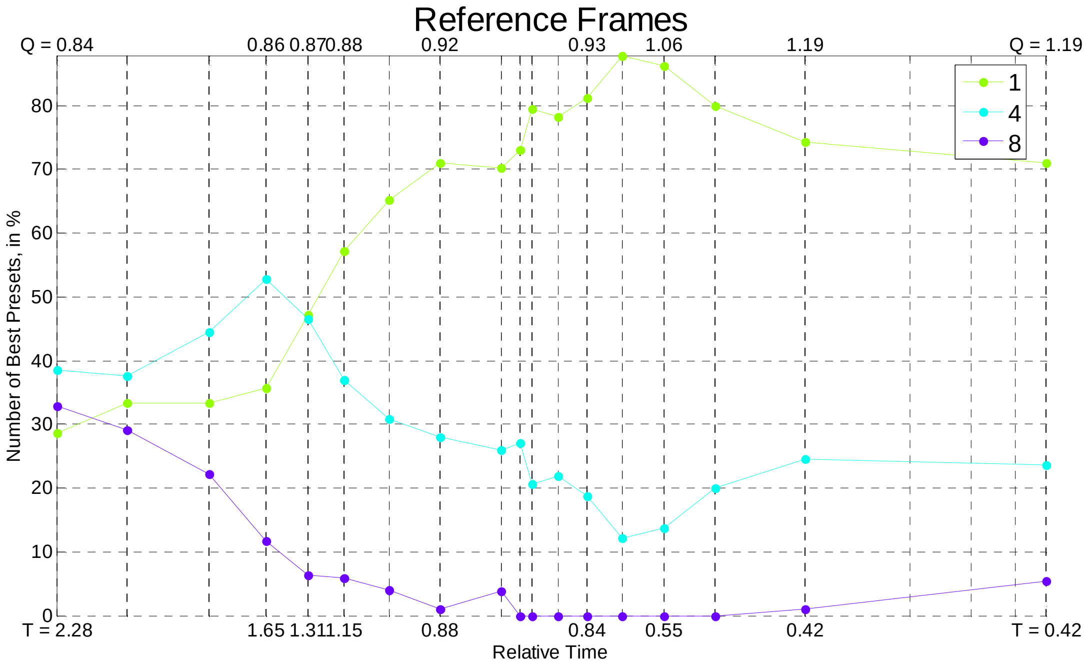
Picture 18. Distribution presets on classes in λ presets analysis (Continuation).



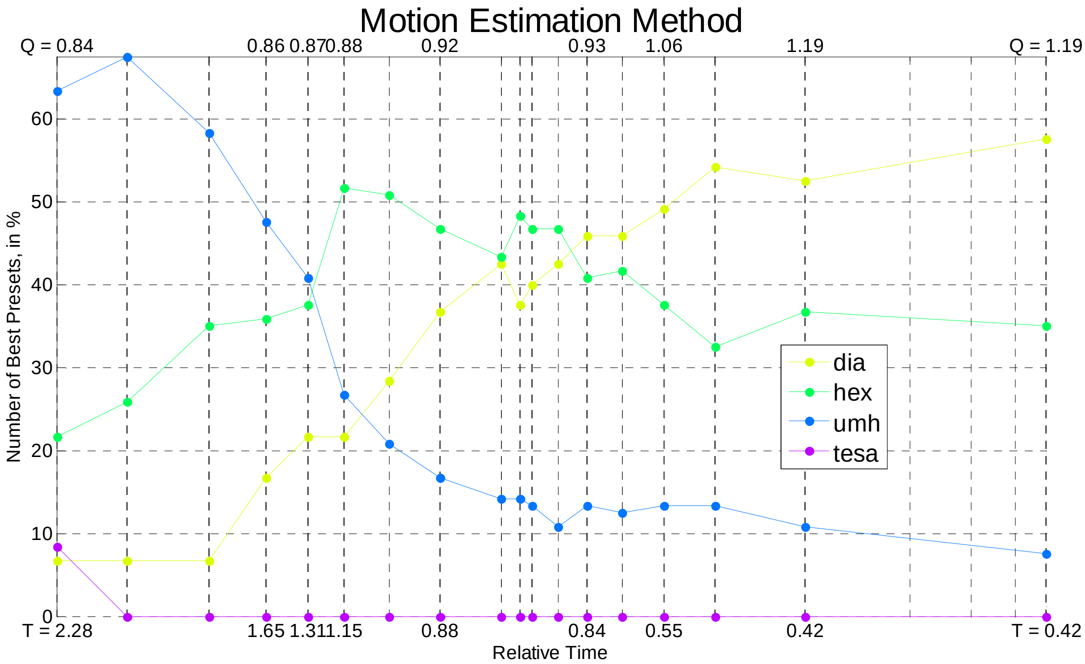
Picture 19. Lambda Presets Analysis of Option Partitions.



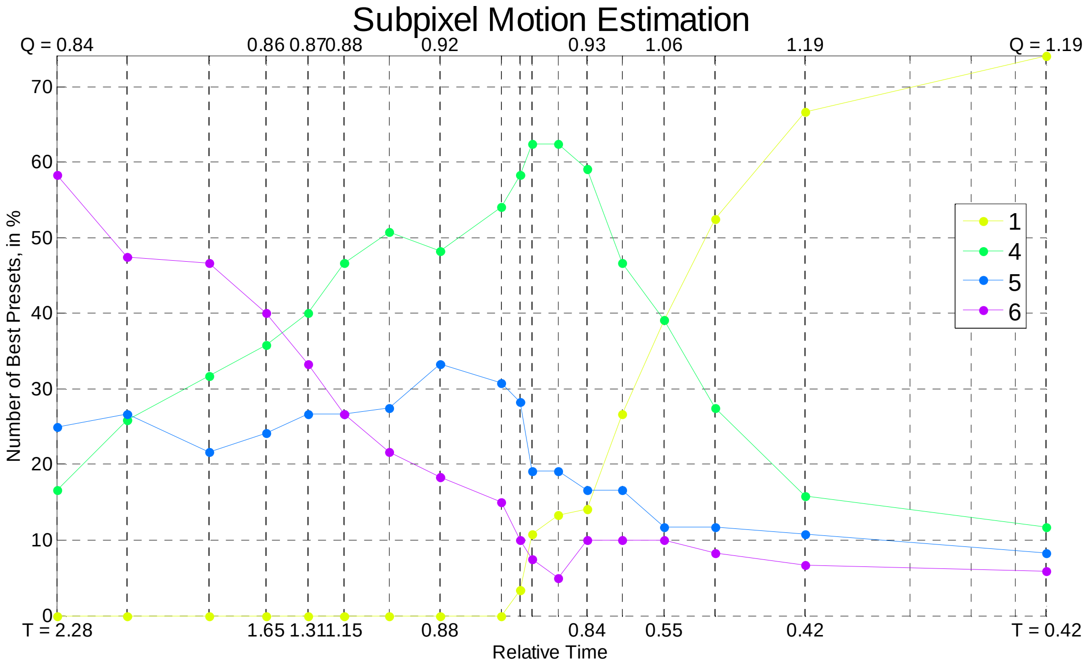
Picture 20. Lambda Presets Analysis of Option B-frames.



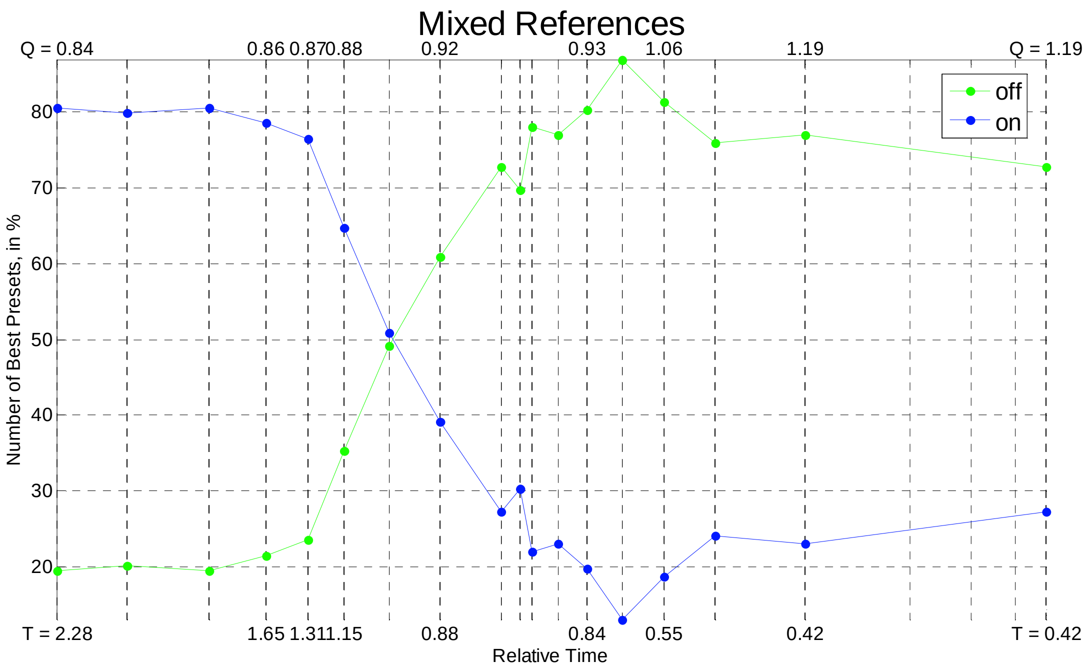
Picture 21. Lambda Presets Analysis of Option Reference Frames.



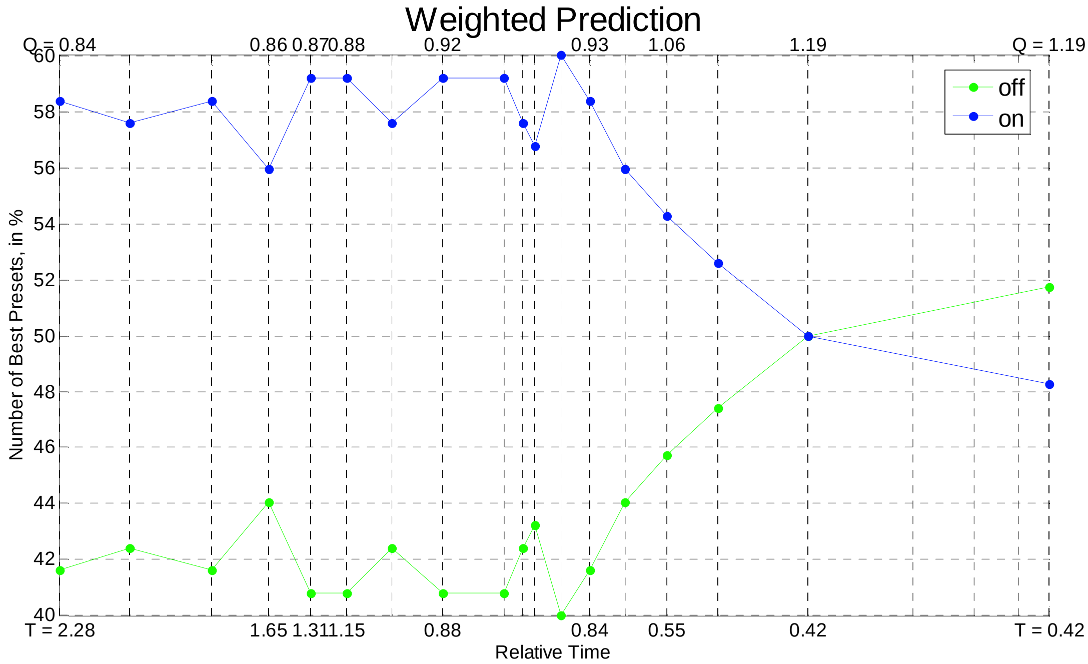
Picture 22. Lambda Presets Analysis of Option Motion Estimation Method.



Picture 23. Lambda Presets Analysis of Option Subpixel Motion Estimation.



Picture 24. Lambda Presets Analysis of Option Mixed References.



Picture 25. Lambda Presets Analysis of Option Weighted Prediction.

Options Analysis

Conclusions from resulted above charts are shown in the table below.

Note that we consider 10% of the best presets for this analysis (see Picture 16). These conclusions can become incorrect if we strongly change this amount.

Any value of λ (ratio of relative encoding time to bitrate, for details see Parameter λ in glossary) correspond just one optimal preset on the chart, i.e. just one optimal time and bitrate. Below we will consider only time value, but imply not only time value, but corresponding bitrate value and value of λ simultaneously.

Table 9. List of Lambda Analysis Results.

Option	Preset	Comments
1. **Partitions** --partitions x	• "none"	If time value greater than 1.31 (31% slower than default preset) the optimal value is "all". If time value belongs to a range [0.85, 1.31] the best value is "p8x8,b8x8,i8x8,i4x4". When time value smaller than 0.85 value "none" should be used. Line "none" has maximum at time 0.55. It means that for the value "none" is better to use when intending to encode in speed approximately twice faster than default x264 preset. For the value "p8x8,b8x8,i8x8,i4x4" the optimal time approximately equals to 0.88. Line according to a value "all" has maximum at time 2.28 (extreme value). It means that for value "all" is more preferable that the time will be greater.
	• "all"	
	• p8x8,b8x8,i8x8,i4x4"	
2. **B-Frames** --bframes n	• 0	Presets without B-frames are not optimal at a chosen 10% of the best presets. Usage of 2 and 4 B-frames does not significantly differ but 2 B-frames are slightly better. It is more preferable for value 0 that the encoding time will be smaller. With the value "2" and "4" is
	• 2	

	• 4	better to use at encoding time approximately equals to 0.88.
3. **Reference Frames** --ref n	• 1 • 4 • 8	Presets with 8 reference frames are not optimal at a chosen 10% of the best presets. If time is lower 1.31 than the best value is “1”. Else the value “4” is more preferable. Value “1” should be used at encoding time approximately equals to 0.84. The optimal encoding time for 4 reference frames approximately equals to 1.65.
4. **Motion Estimation Method** --me x	• “dia” • “hex” • “umh” • “tesa”	Presets with “tesa” ME algorithm are not optimal at a chosen 10% of the best presets. The optimal value at time value greater than 1.31 is “umh”. If time belongs to a range [0.85, 1.31] the best value is “hex”. When time value smaller than 0.85 value “dia” should be used. It is more preferable for value “dia” that the speed will be greater. With the value “hex” is better to use at an encoding time approximately equals to 1.15. It is more preferable for values “umh” and “tesa” than the speed is as small as possible.
5. **Subpixel Motion Estimation** --subme n	• 1 • 4 • 5 • 6	Presets with subme 5 are not optimal at a chosen 10% of the best presets. At time value greater 1.31 the optimal value is 6. If time belongs to a range [0.55, 1.31] the best value is 4. When time value is smaller than 0.55 the value 1 is more preferable. It is more preferable for value 1 that the speed will be greater. With the values 4 and 5 are better to use at an encoding time approximately equals to 0.55. It is more preferable for value 6 than the speed will smaller.
6. **Mixed References** --mixed-refs	• off • on	At time value greater 1.15 the optimal value is “on”. Else the value “off” is more preferable. For the value “off” the optimal encoding time approximately equals to 0.84. It is more preferable for value “on” than the speed will be smaller.
7. **Weighted Prediction** --weightb	• off • on	At time value greater 0.42 the optimal value is “on”. Else the value “off” is slightly more preferable. Values “on” and “off” does not significantly differ at time value smaller 0.42. It is slightly more preferable for value “off” than the speed will be higher. For the value “on” the optimal encoding time approximately equals to 0.88.

Summary

- Using 0 B-frames, 8 reference frames, “tesa” ME algorithm and subme value 5 are not optimal at a chosen 10% of the best presets.
- Usage of 2 and 4 B-frames does not significantly differ but 2 B-frames are slightly better.
- Weighted prediction does not significantly differ quality end encoding time.
- Results of analysis based on lambda show the following tables.

Table 10. List of Lambda Analysis Summary.

	Options						
Time		**> 1.31**	**[1.15; 1.31]**	**[0.85; 1.15]**	**[0.55; 0.85]**	**[0.42; 0.55]**	**< 0.42**
Bitrate		**< 0.87**	**[0.87; 0.88]**	**[0.88; 0.92]**	**[0.92; 1.06]**	**[1.06; 1.19]**	**> 1.19**
λ		**< 0.1**	**[0.05; 0.07]**	**[0.07, 0.4]**	**[0.4, 0.56]**	**[0.56, 5]**	**> 5**
	--partitions	"all"	"p8x8,b8x8,i8x8,i4x4"	"p8x8,b8x8,i8x8,i4x4"	"none"	"none"	"none"
	--bframes	2	2	2	2	2	0
	--ref	4	1	1	1	1	1
	--me	"umh"	"hex"	"hex"	"dia"	"dia"	"dia"
	--subme	6	4	4	4	1	1
	--mixed-refs	on	on	off	off	off	off

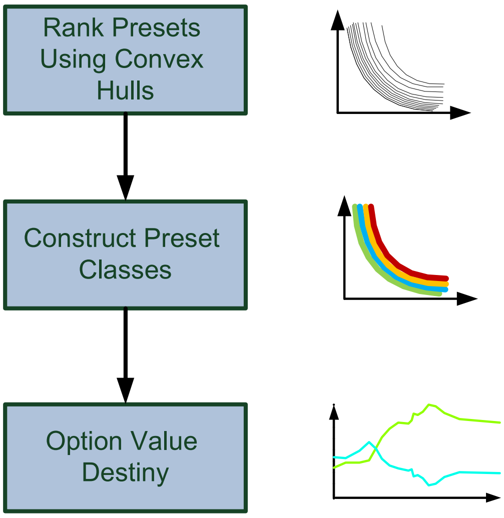
Table 11. List of Lambda Analysis Options Extremes.

	Options						
Time		**min**	**0.55**	**0.87**	**1.15**	**1.65**	**max**
Bitrate		**max**	**1.06**	**0.92**	**0.88**	**0.86**	**min**
	--partitions		"none"	"p8x8,b8x8, i8x8,i4x4"			"all"
	--bframes	0		2			8 (4 same)
	--ref			1		4	8
	--me	"dia"			"hex"		"umh", "tesa" ("umh" better)
	--subme	1		4, 5 (4 better)			6
	--mixed-refs			off			on
	--weightb	off (on same)		on			

Analysis of Distance from Convex Hull

Method Description

The main problem in the options analysis and option values analysis is how to compare presets. As were mentioned earlier best presets have smaller abscissa (relative encoding time coordinate) and smaller ordinate (bitrate coordinate). Thus the closer preset to the left lower corner – the better it is. If we fix time or quality then optimal (for these limits) presets would be lying on the convex hull (or envelope line). See Picture 4.



In this section we analyze presets using this method and the following algorithm. First of all we all presets are ranked using convex hulls discarding. After that, ranking become consolidated and construct presets classes. At the last step we use destiny of presets with option value to analyze current option for selected presets common quality. All steps are described below in derails.

Rank presets using convex hulls

We can say that presets lying on the envelope line (first one) are better than the others, but the number of the others presets is too large and we can't compare these presets among themselves. Let's suppose that there are no dedicated earlier best preset among all our presets. Then we can repeat the process to separate all presets to lying on the convex hull (second one) and the others. It means that if there are no presets dedicated as the best in the first time than presets chosen as the best in the second time were the best. That is presets lying on the convex hull better than the envelope line presets given after casting-out first ones. And the last ones are better than the others presets. Continue further in a similar way we construct the method which allows comparing majority pair of presets.

Construct presets classes

Picture 26 and Picture 27 illustrate the method described above. There are presets colored in the same way correspond to the presets with the same common quality (it means that presets evaluated for speed and quality simultaneous). For convenience we put presets lying on the first 10 convex hulls have the best common quality, presets on the next 10 envelope lines have worse common quality and etc, i.e. we separate all convex hulls into several classes of 10 neighbor convex hulls. Note that the number of presets in classes is different (see Picture 28).

Then we analyze how many presets with fixed value of some option belong to any class of presets with the same common quality. Thus the value of option is better than more presets with this option value belong to first class of presets (presets on first 10 convex hulls) and less presets lying in the last classes of presets (presets on last convex hulls).

Analyzing option value density

The chart for the each option is created in the following way. There is one line corresponding to each option value on the chart. For each class (set of presets with the same common quality, i.e. presets colored in the same way on the Picture 26 and Picture 27) and for each option value we consider the quantity $N(n,k)$ equals to ratio between number of presets with the same common quality (belong to one class n, i.e. lying on convex hulls with numbers $10*n+1,\dots,11*n$) and specified option value k and total number of presets with this option value k:

$$N(n,k) = \frac{N_k^n}{N_k}, \quad (3)$$

where N_k^n – number of presets with the same common quality (belong to one class n) and specified option value k, N_k – total number of presets with this option value k. Then we divide this quantity $N(n,k)$ by sum of quantities $N(n,k)$ for all possible for this option values m (if the total number of presets with this option value k is the same for all possible k then this sum is equal to number of presets in class n) and multiply by 100 to get quantity in percents:

$$N_{\%}(n,k) = 100 \frac{N(n,k)}{\sum_m N(n,m)} \quad (4)$$

This quantity $N_{\%}(n,k)$ corresponds to the point on the chart with class equals to n and belongs to a line corresponds to the option value k. This line consists of such points for all classes.

Some combinations of option values are invalid, for example weighted prediction equals to "on" and b-frames equals to 0. That's why the number of presets with the different values of the same option is various. Therefore we divide the number N_k^n of presets with the same common quality (belong to one class n) and specified option value k by total number N_k of presets with option value *k.* See Formula 3.

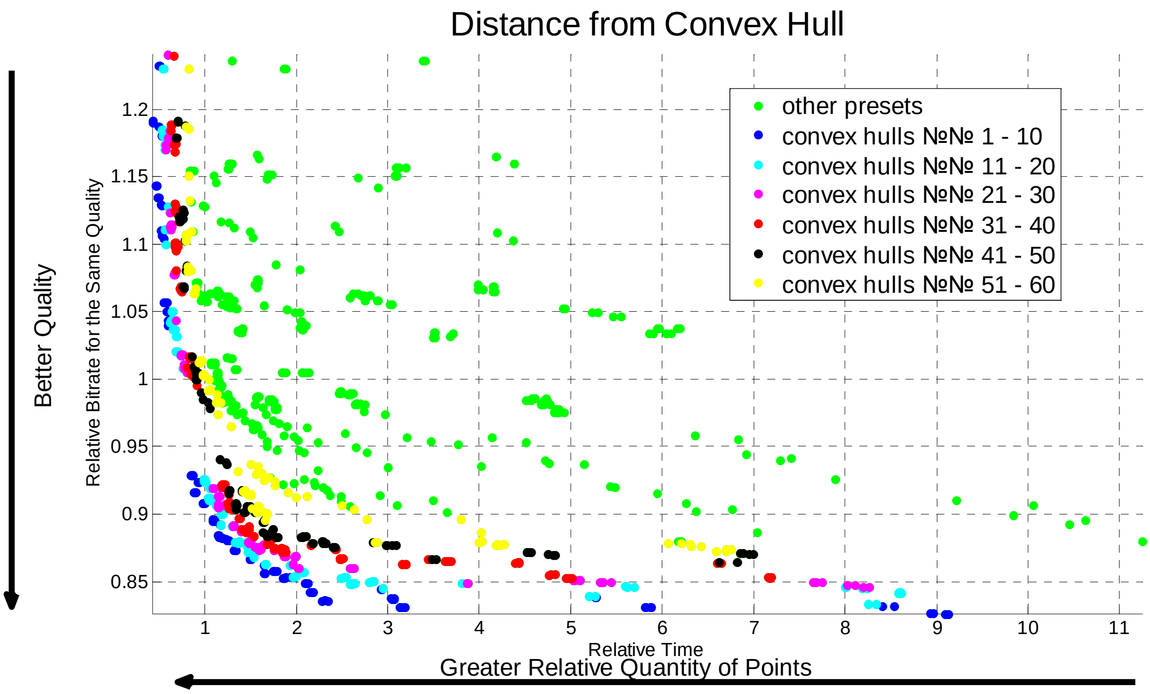
There is the sum of all points with the same X Axis is equal to 100% in the chart according to the definition. See Formula 4.

Thus the line corresponding to the option value is higher at the small values of convex hulls (small number of class) and lowers at the large values of convex hulls (big number of class) – the more preferable this option value. Thus if two lines are intersect each other then we can say that one of them are better than another.

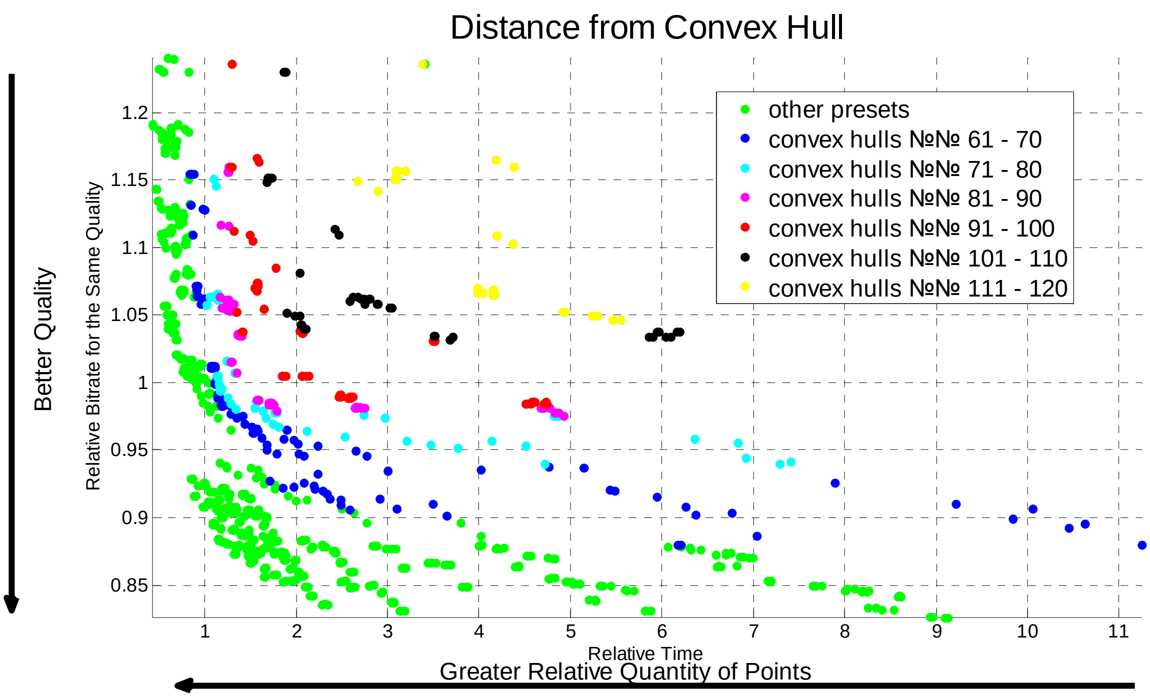
Note that the number of presets in classes is different (see Picture 28) and it is twice smaller in classes 7–11 than in classes 1–6. Therefore contribution of conclusions related to the classes 7–11 would be smaller than conclusions related to the classes 1–6.

Charts for all concerned options are shown below (Picture 29 – Picture 35). Note that the scale of Y axis is varying from chart to chart.

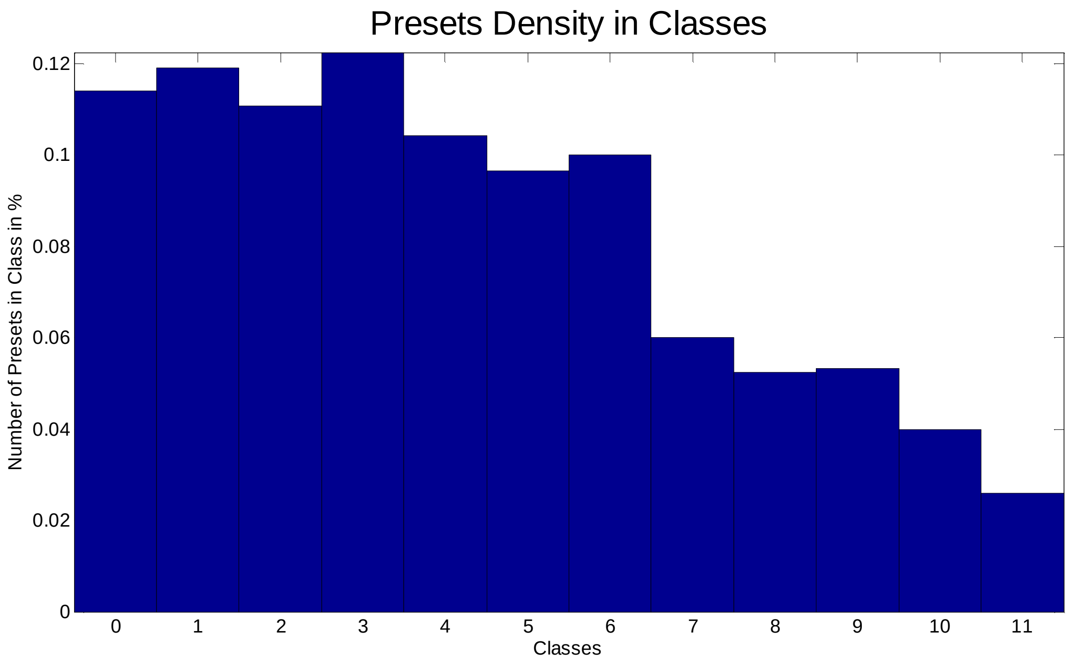
Results



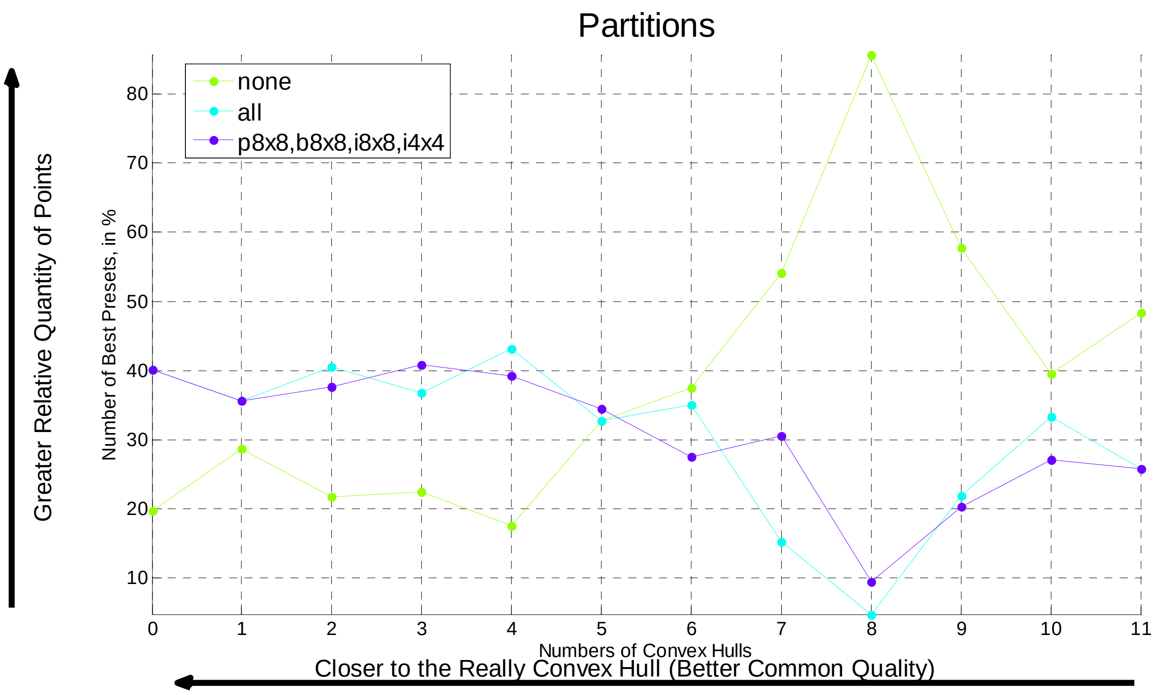
Picture 26. Distribution presets on classes in analysis of distance from convex hull.



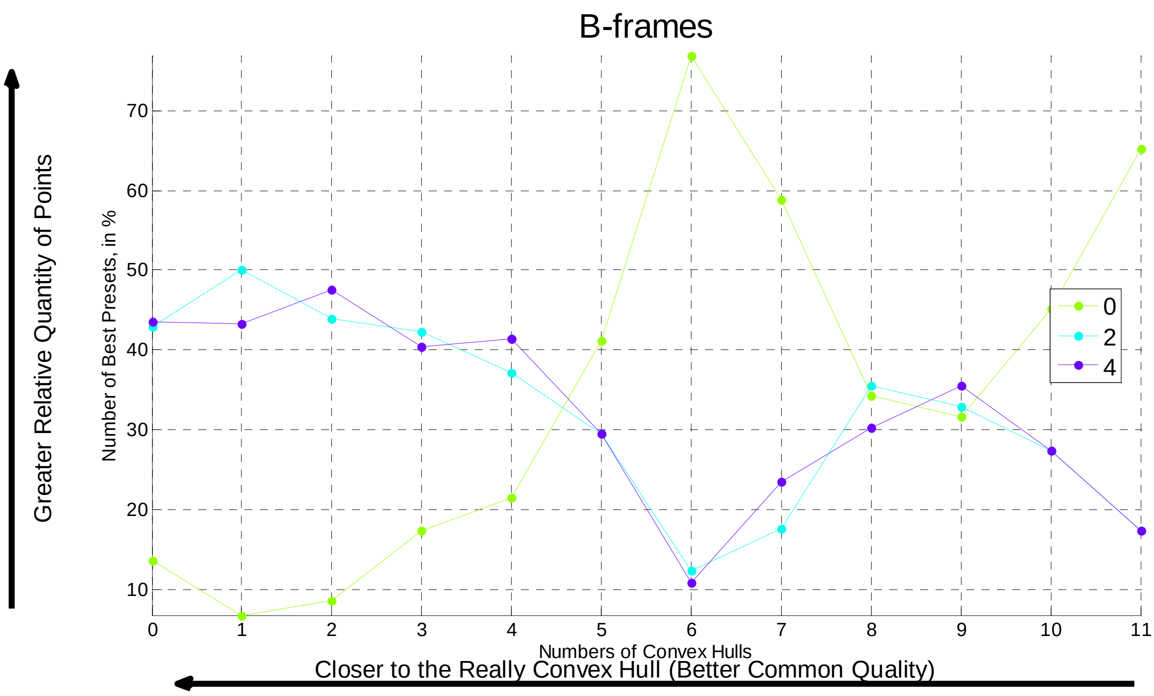
Picture 27. Distribution presets on classes in analysis of distance from convex hull (Continuation).



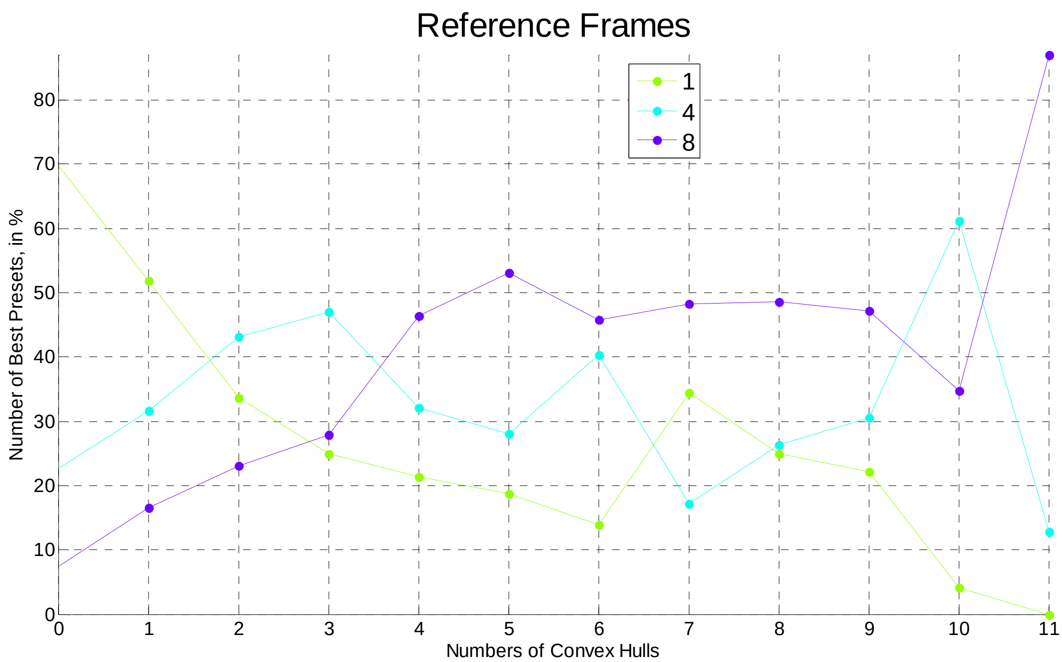
Picture 28. Number of presets in classes in analysis of distance from convex hull.



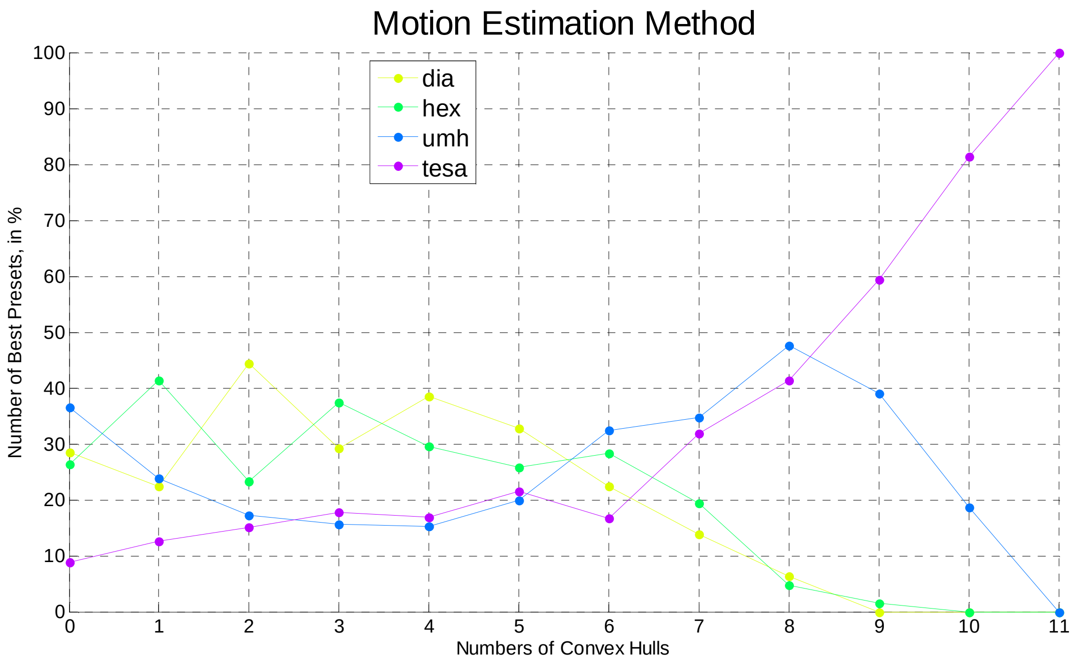
Picture 29. Analysis of Distance from Convex Hull of Option Partitions.



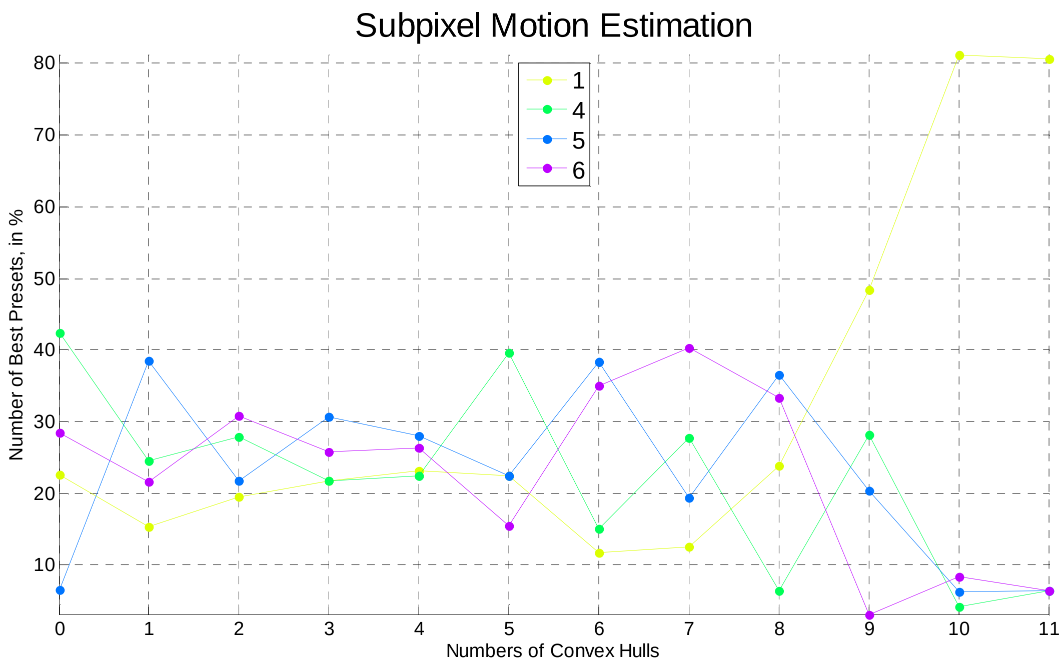
Picture 30. Analysis of Distance from Convex Hull of Option B-frames.



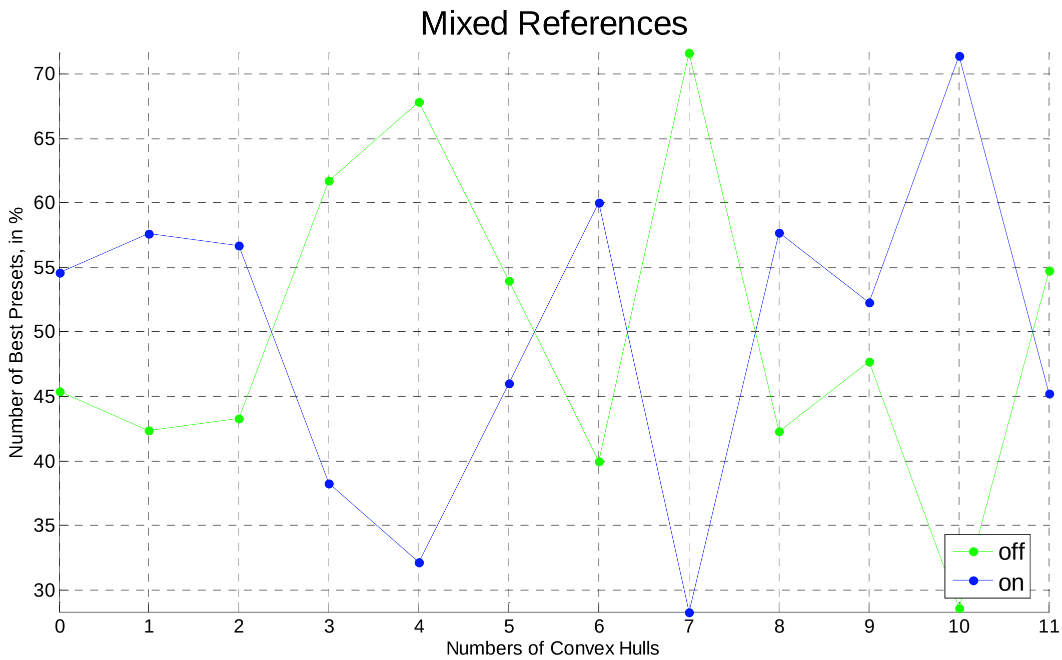
Picture 31. Analysis of Distance from Convex Hull of Option Reference Frames.



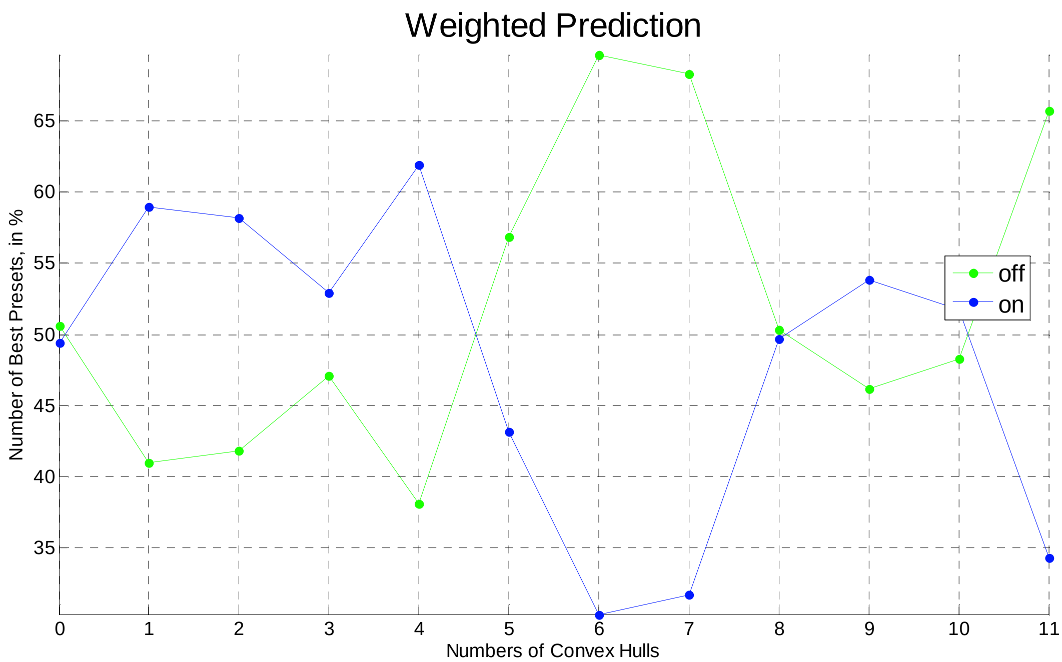
Picture 32. Analysis of Distance from Convex Hull of Option Motion Estimation Method.



Picture 33. Analysis of Distance from Convex Hull of Option Subpixel Motion Estimation.



Picture 34. Analysis of Distance from Convex Hull of Option Mixed References.



Picture 35. Analysis of Distance from Convex Hull of Option Weighted Prediction.

Options Analysis

Conclusions from above charts are shown in the following table.

Note that we analyze some good presets in general. These conclusions can become incorrect if we will consider good presets at specified speed or quality. Fox example, we get that "tesa" algorithm is the worst one (see Picture 32), but if we want to consider presets with the highest quality we will find that "tesa" algorithm is optimal (see Picture 11).

Table 12. List of Analysis of Distance from Convex Hull.

	Option	Preset	Comments
1	**Partitions** --partitions x	• "none" • "p8x8,b8x8,i8x8,i4x4" • "all"	Partitions "p8x8,b8x8,i8x8,i4x4" and "all" are not significantly differing. Presets with partitions "none" are not optimal.
2	**B-Frames** --bframes n	• 0 • 1 • 2	Presets without B-frames are not optimal. Presets with 2 and 4 B-frames are not significantly differing.
3	**Reference Frames** --ref n	• 1 • 4	Presets with 1 reference frame is better than presets with 4 reference frames which is better than presets with 8 reference frames.
4	**Motion Estimation Method** --me x	• "dia" • "hex" • "umh" • "tesa"	Presets with "dia" and "hex" algorithms work better than presets with the other ME algorithms. Presets with "hex" algorithm are slightly better than presets with "dia" algorithm. "umh" algorithm is significantly better than "tesa". But there is the largest number of presets with motion estimation "umh" among the best presets (first 10 convex huls).
5	**Subpixel Motion Estimation** --subme n	• 1 • 4 • 5 • 6	Presets with subme 1 are optimal. Presets with subme 4 are slightly better than presets with subme 6 which works better than presets with subme 5.
6	**Mixed References** --mixed-refs	• off • on	Basically it is possible to tell that presets with "on" value of option mixed references are better than presets with "off" value. But there is large number of presets with mixed references "on" among the worst presets (last 20 convex hulls). So, other options are extremely important to make decision about mixed reference usage.
7	**Weighted Prediction** --weightb	• off • on	Presets with weighted prediction are better than without it.

Summary

- The following option values have the highest density near the true convex hull (i.e. high density among the best presets):
  - --partitions "p8x8,b8x8,i8x8,i4x4", "all"
  - --bframes 2, 4
  - --ref 1
  - --me "hex", "dia", "umh"
  - --subme 4
  - --mixed-refs on

  - --weightb on

- Values “p8x8,b8x8,i8x8,i4x4” and “all” of option partitions do not differ significantly.
- Usage of 2 and 4 B-frames options do not significantly differ.
- Values “dia” and “hex” of option motion estimation do not significantly differ, but “hex” works slightly better.
- Values 4 and 6 of option subme do not significantly differ, but 4 works slightly better.

Analysis of Several Sequences

Method Description

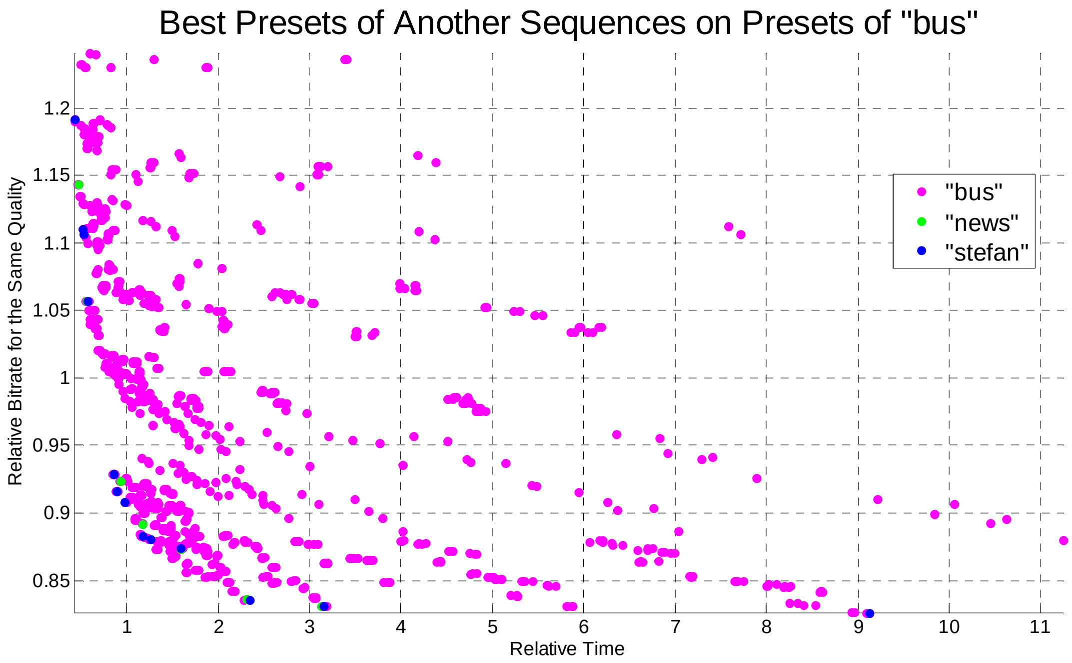
We can't test presets on all sequences, available all over the world. It is very consuming time task even if we have only several sequences. It is desirable to analyze presets only on one sequence and use the result to another sequence.

The goal of this section is to demonstrate the legality of approach mentioned above. We run codec on the three standard different sequences: "bus", "news" and "stefan". For each sequence we find where presets corresponding to other sequences with respect to presets of the current sequence lie.

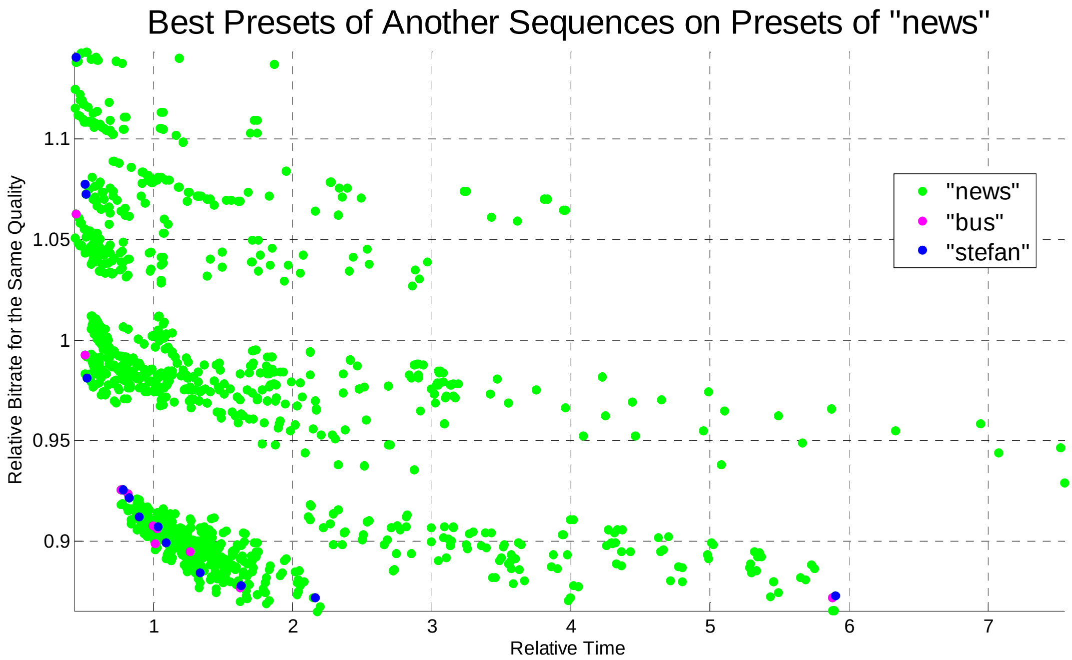
The following charts have been constructed as follows. Consider presets cloud for one sequence. All presets corresponding to the current sequence draw on this chart. For each other sequence convex hull points for this sequence highlighted in the different for each other sequence colors.

Underlying charts show that best presets don't strong dependence of sequence. Best presets for one sequence are closer to other sequences best presets.

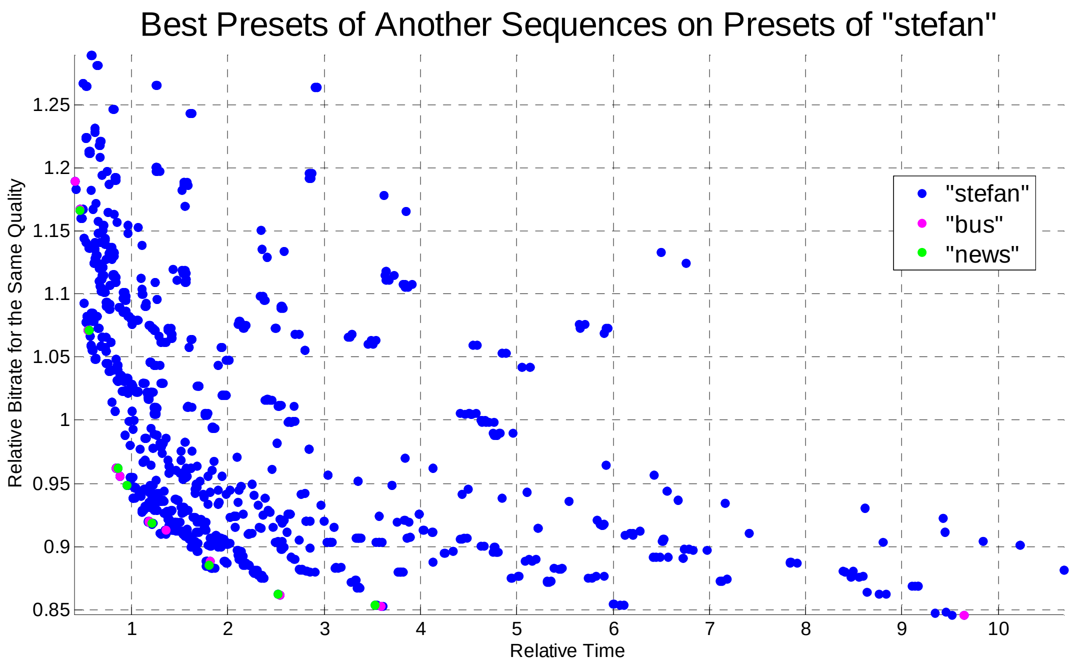
Results



Picture 36. Best Presets of Other Sequences in "bus".



Picture 37. Best Presets of Other Sequences in "news".



Picture 38. Best Presets of Other Sequences in "stefan".

Summary

Best presets for one sequence are closer to best presets for another sequence. So we can analyze best presets only for one sequence and the results will be more or less correct to another. We considered only one test sequence "bus" to analyze quality of presets and corresponding options above and now we substantiate this approach.

Conclusions

There are four methods of video codec options analysis were used in this report:

- Best presets selection of the codec using presets convex hull. See Table 4 and Table 5 for more details.
- Options analysis using colored clouds of points. This method is very easy to use, but some subjective interpretation is possible.
- Analysis of different speed/quality tradeoff using Lambda parameter. This type of analysis allows to make some conclusions about option efficiency for different speed/quality tradeoff.
- Analysis using convex hull deletion. This analysis separate preset to "good" and "bad" relative to other presets without any division to high speed and high quality presets.

Future Plans – Possible Analysis Methods

There are number of possible research directions for options analysis:

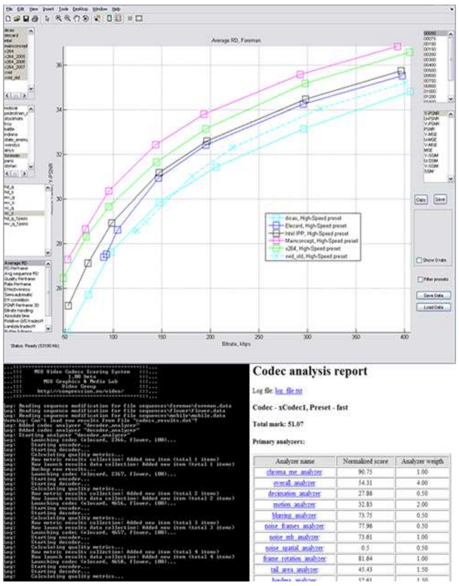
- Options dependence. It is common situation when *"Option X should be used when option Y have value Z"*. Our methods are not suited to track such relationships.
- Methods verification:
    - Encoding speed confidence interval;
    - Quality estimation confidence interval (errors of RD curve approximation);
    - Stability of results for different sequences.
- Stability of each option for different sequences. Is it possible to make any conclusions for an option without taking into account video sequence parameters?

List of Pictures

List of Tables

This comparison was performed with ViCoS – Video Codec Scoring System

About the Video Codec Scoring System



ViCoS is a fully automatic quality evaluation system for video codecs and video processing algorithms.

It is an advanced system with client-server architecture and relational data base support. It allows robust codec launches with user-friendly interface and functions for video codec or video filter analysis with easy-to-use visualizations of results. With ViCoS you can:

1. **Perform QA with much lesser resources**
   ViCoS usage allows to do Quality Assurance tasks in a highly automatic way. Now video codec features or entire codec quality can be tested very easily without big number of QA specialists.
2. **Perform codec testing without subjective codec testing**
   ViCoS implements many different quality analyzers that can replace expensive subjective quality evaluation for almost every task.
3. **Fast comparison to competitors**
   ViCoS provides functionality for video codecs comparison. Now codec developers can compare their video codec quality to competitors very fast and easily.
4. **Choose optimal default and predefined parameters**
   ViCoS can help to choose optimal (speed/quality trade-off) encoding parameters using preset analysis subsystem.
5. **Compare different versions of a product easily**
   ViCoS helps to perform quick speed and quality comparison of different versions of a codec or video processing software.

And much more.

Main key features of the system:

1) *Client-server architecture*.
2) *Easy modifications* to add a new codec, preset or video sequence.
3) *Robust launches* – if a codec fails the system continues to work, marking the error for this codec
4) *DB usage* – all results can be saved in a data base (almost any relational data base management systems: MySQL, MSSQL, Oracle, etc.)
5) *Result visualization* – all obtained results can be visualized very quickly with user friendly-interface.
6) *Huge Amount of Data Processing* – during ViCoS work huge amount of data is produced, it is processed and categorized very easily and user friendly.
7) *Specific Analysis Types* – ViCoS uses specific types of analysis: well-known and specially developed (Edge capture, Borders quality, Tail area, Blurring, Synthetic motion, and more than 10 other types).

More information could be found at http://yuvsoft.com/technologies/vicos/

E-mail: vicos@yuvsoft.com

MSU Video Quality Measurement Tool

MSU 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

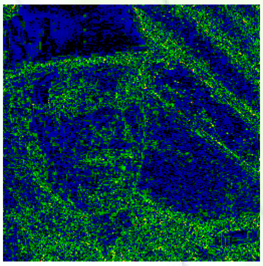
GUI and command line tools

7. Plugins Interface

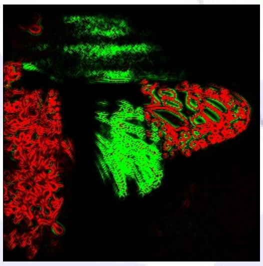
You can easily develop your own metric

Visualization Examples

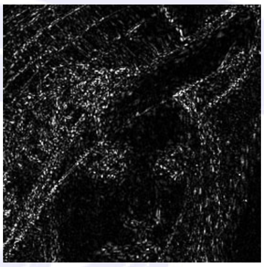
Allows easily detect where codec/filter fails

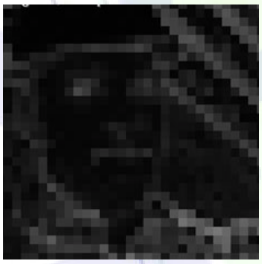

Y-YUV PSNR


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: