ViCoS – Video Codec Scoring System

GUI Description

Moscow, Russian Federation, 2010

phone: +1-(408)-426-5988

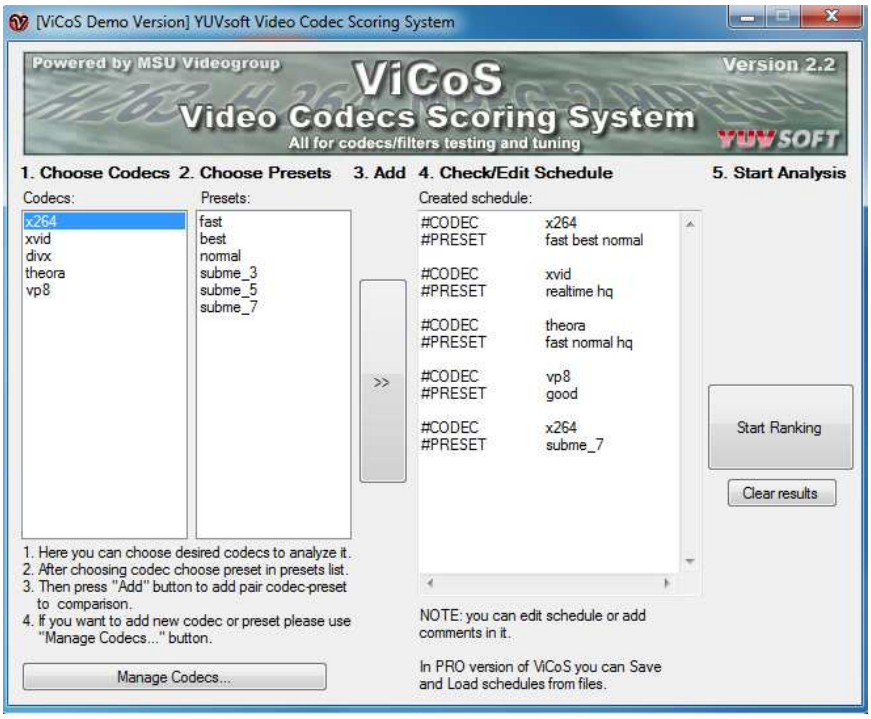
vicos@yuvsoft.com

http://www.vicos.yuvsoft.com/

1 Table of Contents

2 ViCoS GUI Description

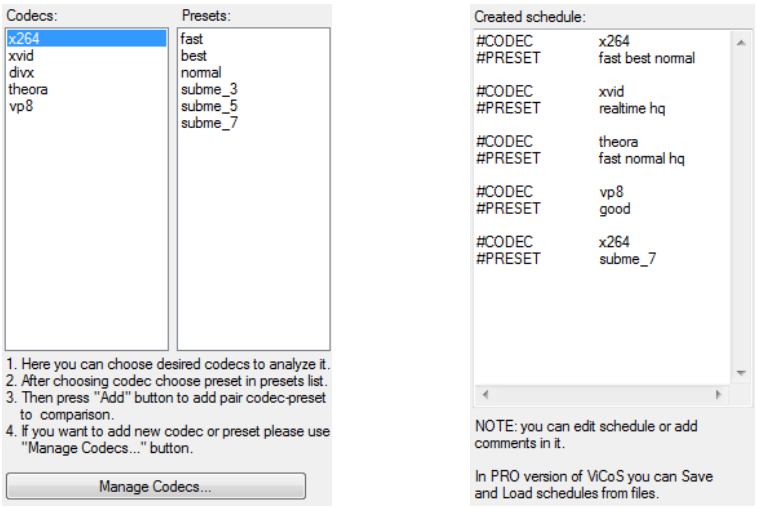
2.1 Main Dialog Window



Pic. 1 The main system window

The main system window consists of three main forms:

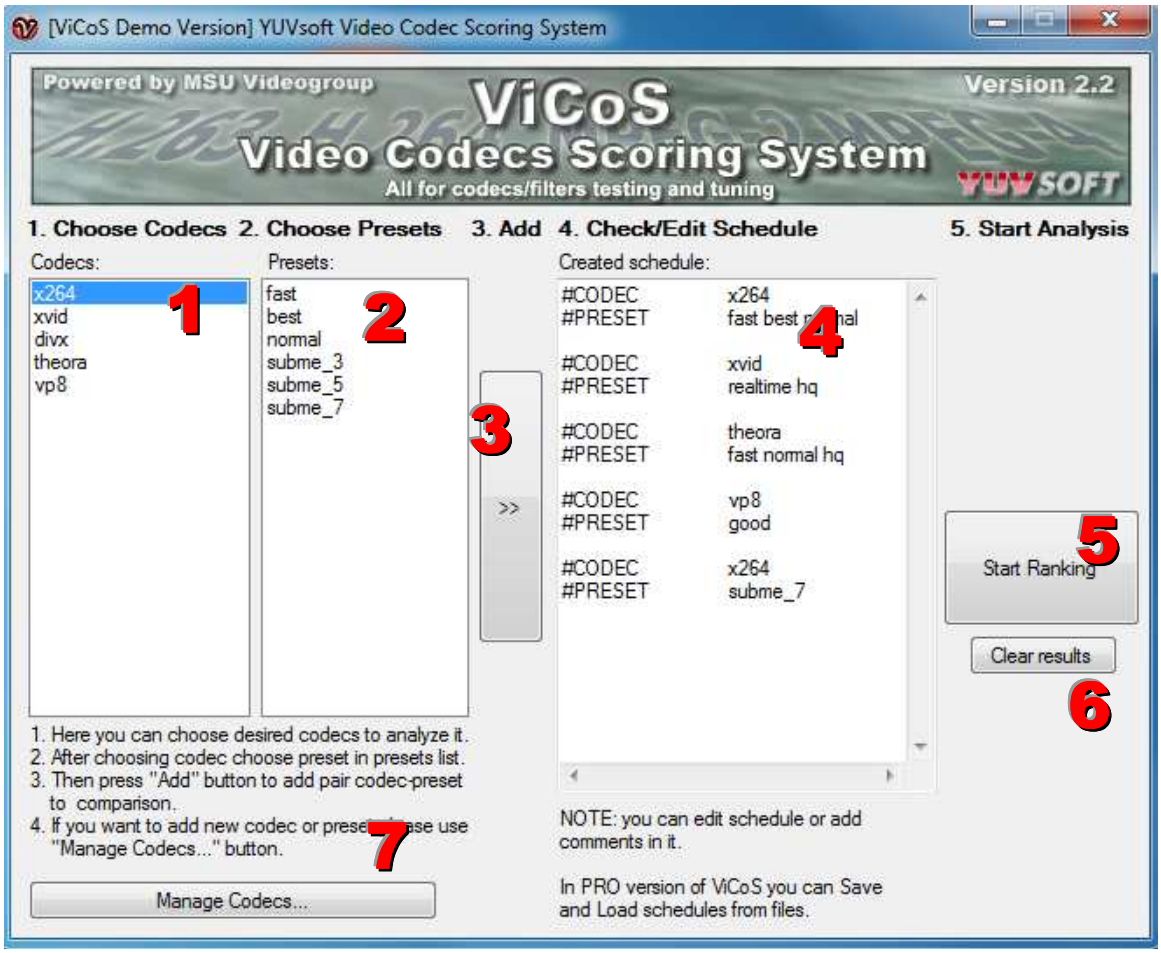
- Codec and preset selection forms
- Schedule



Codec and Preset selection forms Schedule

Pic. 2 Main forms of the main window

Now let's consider all buttons and forms in the main system window. Every button and form is marked with number and described next.



Pic. 3 The main system window with marked buttons

1. **Form “Codecs”**
   In *Codecs* window user can choose desired codecs for testing. Corresponding presets will appear in *Preset* window after choosing codec. To select a codec click on it with mouse left-button.

2. **Form “Presets”**
   This form contains presets available for chosen codec in *Codecs* window. To select a preset click on it with mouse left-button.

3. **Add button “>>”**
   Pressing Add (>>) button will bring codec-preset pair to *Created Schedule* form.

4. **“Created Schedule” Form**
   This form contains whole schedule for testing. User can manually edit it in this form. Schedule consists of simple units, like this:

   #CODEC x264
   #PRESET fast default quality

   First row *#CODEC* contains codec names for testing, second row *#PRESET* contains presets for this codec for testing. User can merge or split units. The next units are equal:

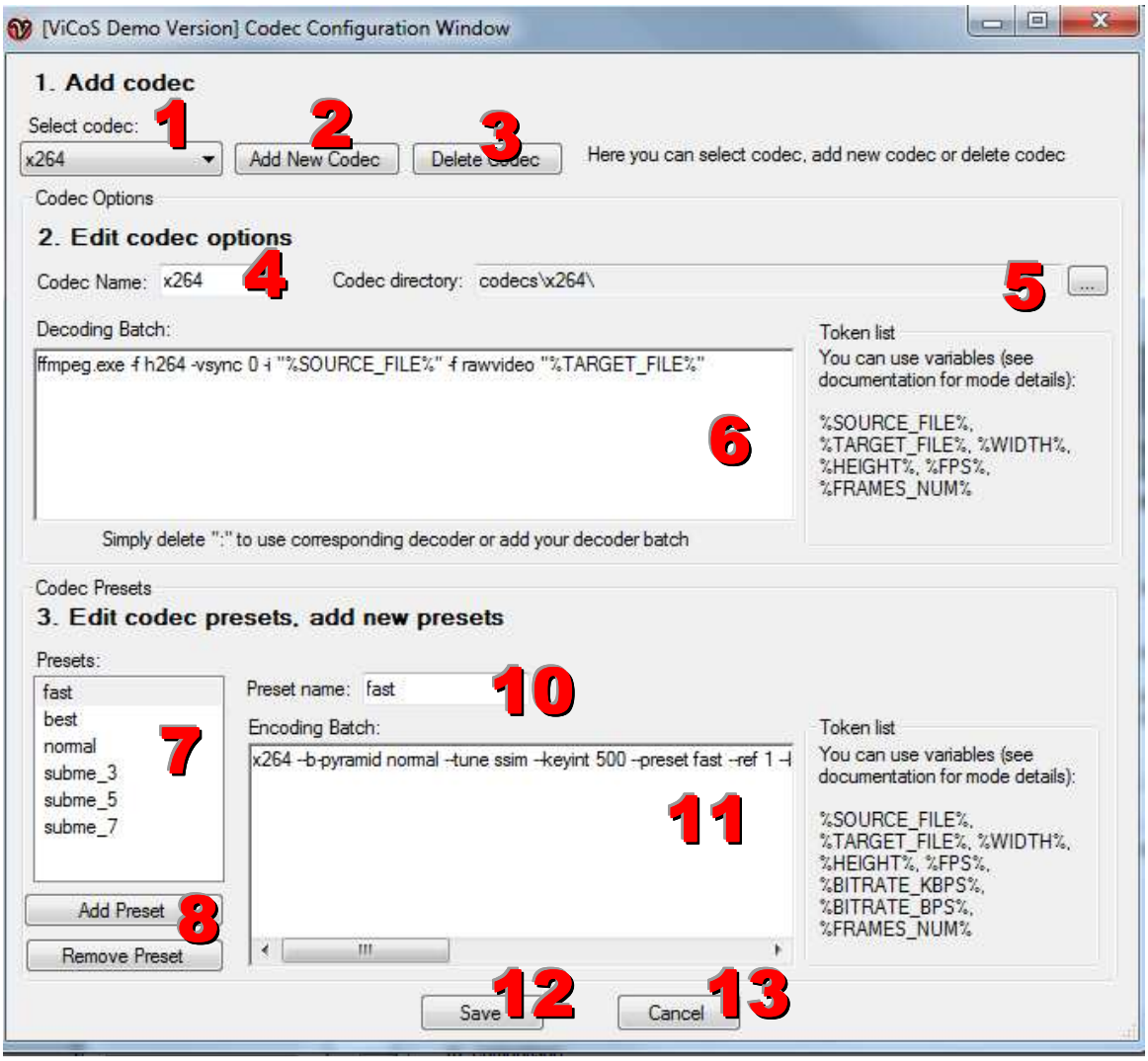
   #CODEC x264
   #PRESET fast default quality

   #CODEC x264
   #PRESET fast default

   #CODEC x264
   #PRESET quality

5. **“Start Ranking” button**
   After creating schedule user can start codec testing with schedule by pressing this button. After system starts Run System Window will appear.
6. **"Clear Results" button**
   This button clear all calculated results for all codecs and presets.
7. **“Manage Codecs …” button**
   This button brings Codec Configuration Window for editing codecs and presets: user can delete, add, or edit codecs and presets.

2.2 Codec Configuration Window



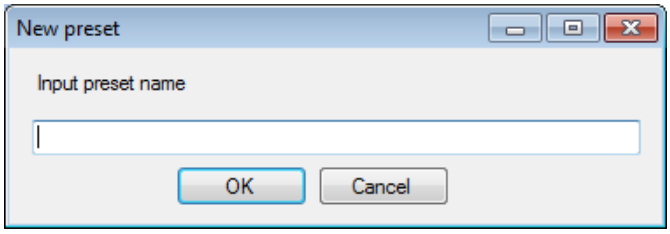
Pic. 4 The codec configuration window with marked buttons

This window consists of three main parts:

- Add codec - for adding or removing codecs.
- Edit codec options - for editing codec options and decoding batch.
- Edit codec presets, add new presets - for editing preseta or adding new presets.

This window is intended for codec and presets configuration and description.

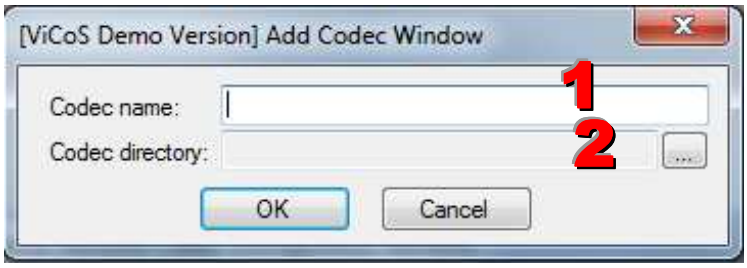
1. **“Select codec” Combo-box**
   Here user can choose codec described in the ViCoS.
2. **“Add new Codec” button**
   User can add new codec with this button. After pressing it Add codec window will appear.
3. **“Delete Codec” button**
   This button deletes codec from the ViCoS codec list.
4. **“Codec Name” form**
   Here user can input name for codec – this name will be used in ViCoS.
5. **“Codec directory” form**
   It is a folder that contains executable file for codec, and the batch files for encoder and decoder start should be used this folder as base folder for path settings.
6. **“Decoding Batch” form**
   In this form user can edit decoding batch for chosen codec. User can use different variables, described in *Token list* left to form.
7. **“Presets” form**
   User can choose preset for chosen codec to edit encoding batch or re-name preset.
8. **“Add Preset” button**
   User can add new preset by pressing this button. After pressing it *New preset* window will appear.



Pic. 5 New preset window

9. **“Remove Preset” button**
   User can remove preset by pressing this button.
10. **“Preset name” form**
    User can rename preset in this form.
11. **“Encoding Batch” from**
    In this form user can edit encoding batch for chosen codec. User can use different variables, described in *Token list* left to form.
12. **"Save" button**
    User can save edited or added preset.
13. **"Cancel" button**
    User can cancel any changes to codec or preset.

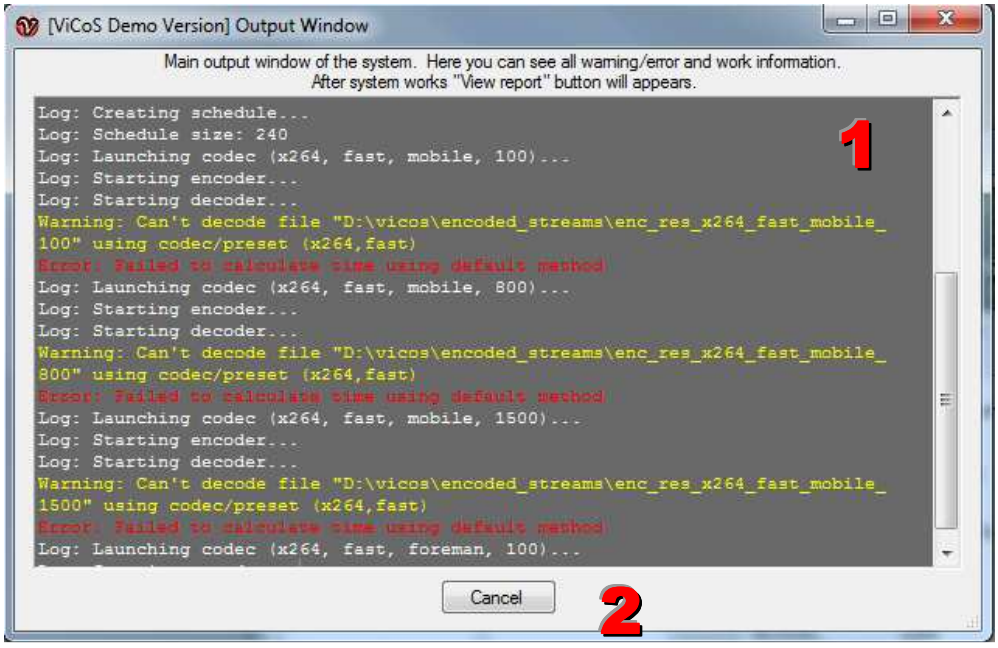
2.3 Add codec window



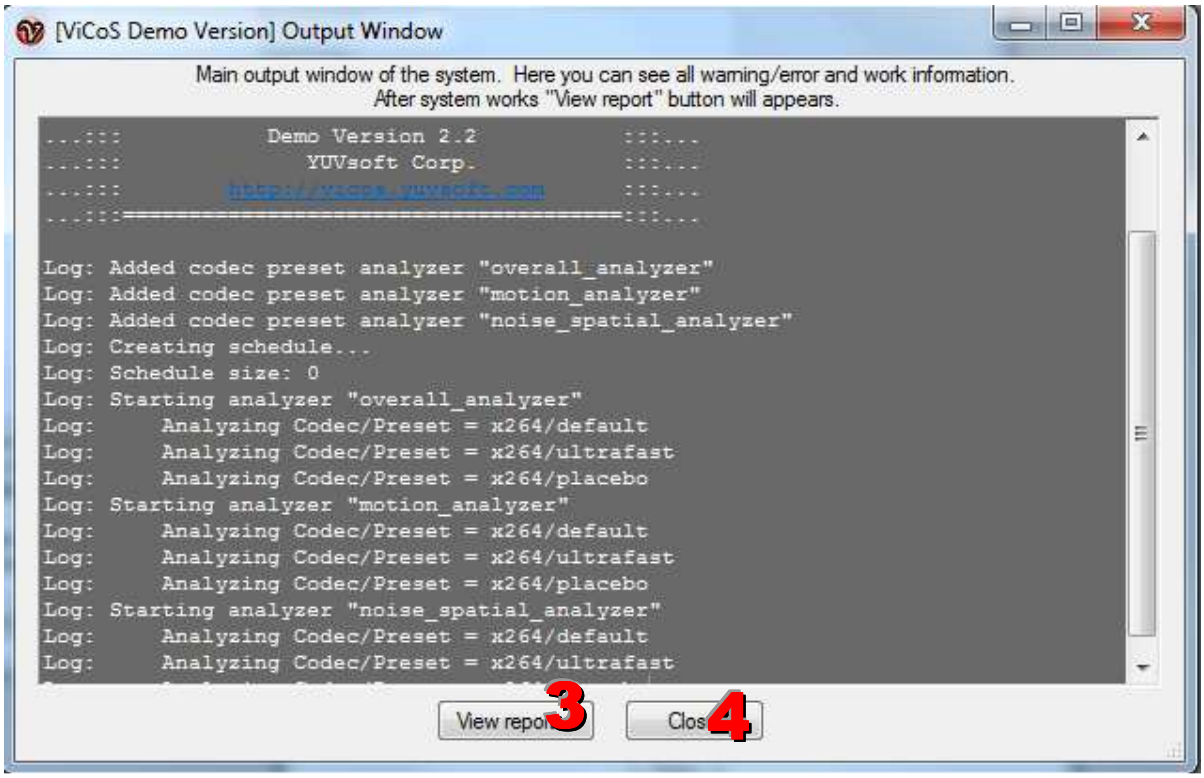
Pic. 6 The add codec window with marked buttons

1. **"Codec name" form**
   User can enter name for the new codec in this form.
2. **"Codec directory" form**
   It is a folder that contains executable file for codec, and the batch files for encoder and decoder start should be used this folder as base folder for path settings.

2.4 "Run System" window

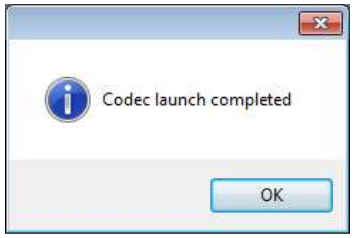


Pic. 7 The run system window during work with marked buttons



Pic. 8 The run system window after work with marked buttons

Run system window during ViCoS work contains main window and cancel button. After system works two buttons (View report and Close) will appear and info window appears.



Pic. 9 The info window

Now let's consider all buttons and forms.

1. **Main output window**
   It is window where ViCoS inputs all information including warnings and errors.
2. **"Cancel" button**
   User can cancel ViCoS work by pressing this button.
3. **"View report…" button**
   After ViCoS work user can view overall report by pressing this button.
4. **"Close" button**
   User can close the Run System window by pressing this button.

3 Editing Codecs Parameters

Each codec, which can be launched using batch commands, can be inserted into ViCoS system. Each codec can be launched using different parameters, so called “presets”. To describe preset, you should write batch commands for your codec.

Following variables can be used in batch command.

Variable name	Description
%BITRATE_BPS%	Target bitrate, bits per second
%BITRATE_KBPS%	Target bitrate, kilobits per second (1 kbps = 1024 bps)
%BITRATE_KBPS1000%	Target bitrate, kilobits per second (1 kbps = 1000 bps)
%SOURCE_FILE_AVS%	Source file, wrapped in AVS file (AviSynth and RawSource plug-in should be installed)
%FRAMES_NUM%	Number of frames in source sequence
%SOURCE_FILE%	Source Raw video in YV12 format
%TARGET_FILE%	File to save encoded video to
%HEIGHT%	Video height, pixels
%WIDTH%	Video width, pixels
%FPS%	Source file framerate, frames per second

Following variables can be used in decoding batch:

Variable name	Description
%FRAMES_NUM%	Number of frames in source sequence
%SOURCE_FILE%	Source raw video in YV12 format
%TARGET_FILE%	File to save encoded video to
%HEIGHT%	Video height, pixels
%WIDTH%	Video width, pixels
%FPS%	Source file frame rate, frames per second

4 Analyzers Description

4.1 Overall Analyzer

The main idea of overall quality analyzer is to simultaneous use number of objective quality metrics to increase subjective quality assessment. Averaging of raw data of quality metrics is not correct because of each of video metrics has its own scale. But we can define scale independent measure for each video metric—bitrate ratio for the same quality.

Non-ideal bitrate keeping for the CBR mode and non-ideal control of overall distortion for encoders don't allow us to compare quality for the same bitrate or bitrates for the same quality. That is the main reason of RD curves usage. Analyzing integral properties of codecs, we should use average bitrate ratio instead of bitrate ratio in point.

The calculation of overall quality score consist of three steps:

- *Calculation of several quality metrics*. Three metrics are used in this analyzer: PSNR and SSIM. Overall quality for all color planes is calculated for each of them.
- *Calculation of average bitrate ratio*. Single reference codec is used for bitrate ratio calculation.
- *Averaging of obtained bitrate ratios*. Linear combination of marks is used. Averaging coefficients are selected based on correlation between results of subjective and objective assessments for the metric. We used our own subjective comparison of modern video codec[1] to get subjective data. Final coefficient for metric is normalized square of Pearson correlation. Table below shows used subjective data and resulting metrics coefficient.

Video Metric	Pearson Correlation to Subjective Data	Metric Coefficient
PSNR	0.802	0.313
SSIM	0.937	0.428

After calculating of averaged mark for the codec, we produce final analyzer mark using the following formula:

$$Score = 1/M$$

[1] MSU Subjective Comparison of Modern Video Codecs, 2006 http://www.compression.ru/video/codec_comparison/subjective_codecs_comparison_en.html

where M is resulting relative marks for codec/preset pair

4.2 Motion Analyzer

Another possible technique to analyze codecs is to use synthetic sequences instead of natural sequences modifications. From one hand, synthetic sequences are not target type of video for codecs. From the other hand, we have much more possibilities to control synthetic sequences, allowing us to influence on codec more effective.

Synthetic sequence is used for detailed analysis of motion compensation algorithm. Sequence consists of squares, which move from frame to frame. Each object can be described with the following parameters:

- Texture
- Size
- Position (X, Y)
- Speed (V_x, V_y)

Synthetic texture is created using the following formula for each color space:

$$C(x,y) = \frac{255}{4}(\sin(fr_x \cdot x) + \sin(fr_y \cdot y) + 2),$$

where fr_x and fr_y are randomly selected frequencies.

Background texture of frame is generated using similar method:

$$C(x,y) = \frac{200}{8}(\sin(fr_{x1} \cdot x) + \sin(fr_{x2} \cdot x) + \sin(fr_{y1} \cdot y) + \sin(fr_{y2} \cdot y) + 4)$$

where fr_i are predefined constants.

Size of each object selects randomly using normal distribution with parameters (*MAX_SIZE*/2, *MAX_SIZE*/8), where constant *MAX_SIZE* depends only on frame size.

Initial position of each square is random. Later for each frame (*i+1*) position calculated using following formula:

$$X_{i+1} = X_i + V_x^i$$
$$Y_{i+1} = Y_i + V_y^i$$

Calculating of the rectangle's speed has two stages:

1. Adding random component to the speed. Uniform random variate in segment [0, *MAX_SPEED*] is used here. *MAX_SPEED* constant is used to control sequence complexity.
2. Calculating correlation component to the speed between. Correlation component (C_x, C_y) for rectangle *k* is calculating using following formulas:

$$(C_x, C_y) = \sum_{n=1}^{N} (C_x^{n,k}, C_y^{n,k})$$

$$C_x^{n,k} = \begin{cases} F_1 \frac{Y_n - Y_k}{d^2}, d < MAX\_DIST \\ F_2 \frac{X_n - X_k}{d^2}, d \geq MAX\_DIST \end{cases},$$

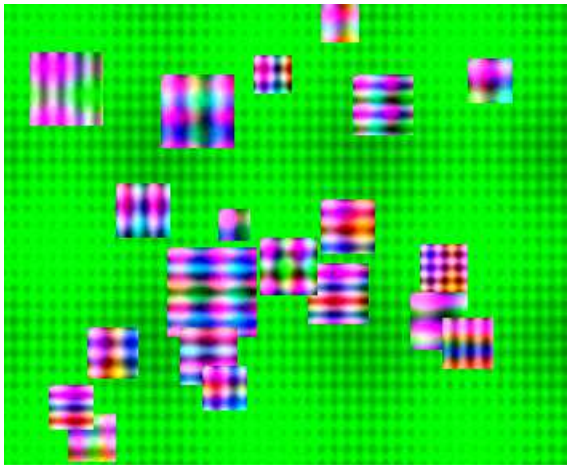
$$C_y^{n,k} = \begin{cases} -F_1 \frac{X_n - X_k}{d^2}, d < MAX\_DIST \\ F_2 \frac{Y_n - Y_k}{d^2}, d \geq MAX\_DIST \end{cases}$$

where *MAX_DIST*, F_1 and F_2 are constants, "*d*" is distance between rectangle "*k*" and "*n*":

$$d = \sqrt{(X_k - X_n)^2 + (Y_k - Y_n)^2}$$

Correlation component used for emulation of correlation between objects motion is scene.

0 demonstrates example of frame from synthetic sequence.



Example of Frame from Synthetic Sequence for Motion Analysis

Instead of complexity sequences, this analyzer uses reference codec for final score estimation.

First step is relative marks calculation. Tested codec in launched for each synthetic sequence, comparing resulting quality to the reference codec. So, for each

sequence and codec, we have one number (average bitrate ratio for the same quality relative to reference codec).

Final score consist of two parts:

- Encoding quality for the first version of motion sequence
- Quality changing for the second version.

Final score fitting function is the following:

$$Score = 100 - \alpha \cdot S_{abs} + (1-\alpha) \cdot S_{ch}$$
$$S_{abs} = M_0$$
$$S_{ch} = M_1 / M_0$$

where M_0 is relative mark for motion sequence with parameter 1

M_1 is relative mark for motion sequence with parameter 15

α is method parameter (currently 0.5 is used)

4.3 Spatial Variable Noise Analyzer

Next type of modifications is adding spatial variable noise to video sequence. Addition of noise to the sequence increases distortion for the same quality (overall encoding complexity), but we are interesting exactly in spatially variable noise.

Normal spatially distributed noise was used in this modification. Noise parameter σ changes linearly from 0 for first pixel of the picture to defined constant *MAX_SIGMA* for last pixel (line-per-line scan order was used for pixels arrangement). *MAX_SIGMA* is main parameter to control complexity of generated sequences. Examples of modified sequences are demonstrated at 0.

Examples of Sequences with Added Spatial Noise

MB-level rate control is the main part of codec, which can be analyzed using produced sequences. Parameters of MBs are changed inside one frame, increasing importance of quantizer selection for each MB. Formula for final mark of this analyzer is

$$Score = 1/M_1$$

where M_1 is synthetic sequence score with *MAX_SIGMA* = 8.

5 Example

In this example, we will add VP8 CLI encoder and compare it with x264.

1. Install YUVSoft ViCoS Demo. You can choose installation directory in installation wizard. Let's suppose, that you installed ViCoS to D:\vicos\ folder.
2. Download and compile, if necessary, VP8 sources. For example, snapshots from http://code.google.com/p/webm/downloads/list can be used. All we need for testing is only two binary files: *ivfenc.exe* (console encoder) and *ivfdec.exe* (decoder).
3. We will place encoder in separate folder, D:\codecs\vp8\. Create this folder and copy files *ivfenc.exe* and *ivfdec.exe.*
4. Start YUVSoft ViCoS Demo GUI using Start Menu or directly by launching D:\vicos\auto_tester_gui.exe.
5. First time you will be suggested to download sequences from the web. Note, that list of sequences is fixed in Demo version.
6. Click "Manage Codecs…" button on the main dialog to open codec management dialog.
7. Press "Add New Codec" button. In appeared dialog enter name of the codec (let use "vp8") and choose codec directory (D:\codecs\vp8\). New codec (vp8) is added to the system
8. Write down decoding batch. For vp8 it is rather simple:

   ```
   ivfdec.exe "%SOURCE_FILE%" -o "%TARGET_FILE%"
   ```

9. Add one or more presets to the codec. Press "Add Preset" button and enter name of the preset: "*good*", for example.
10. Ender encoding batch for the preset:

    ```
    ivfenc.exe --good --token-parts=3 --threads=4 --end-usage=0 --drop-frame=0 --timebase=1000/%FPS%000 --target-bitrate=%BITRATE_KBPS1000% -w %WIDTH% -h %HEIGHT% "%SOURCE_FILE%" "%TARGET_FILE%"
    ```

11. Press "Save" button to close codec management window.
12. Add new codec to the schedule. Select "vp8" in codec's list, "good" preset in presets list and press ">>" button. Following test should be appended to the schedule:

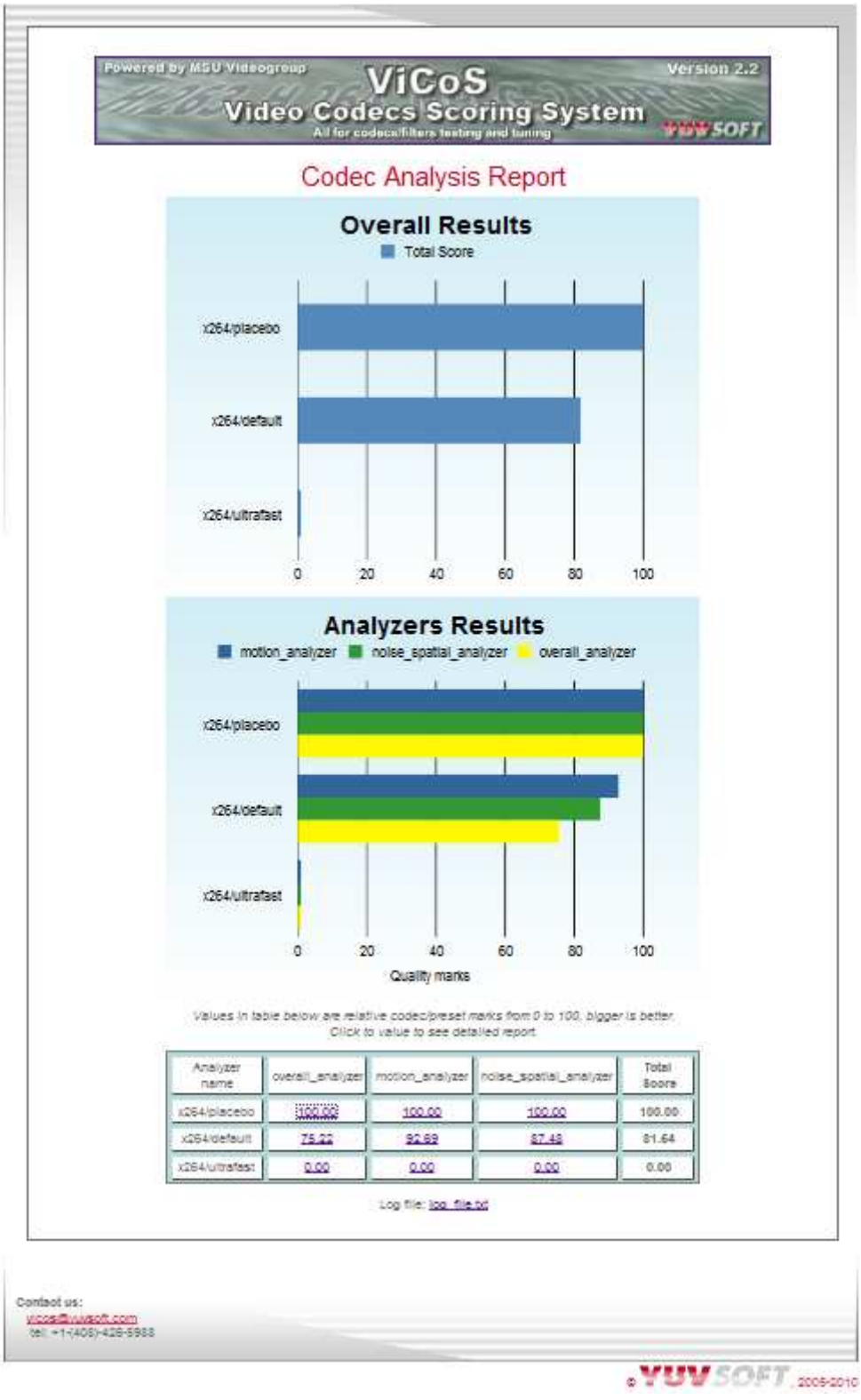
    ```
    #CODEC  vp8

    #PRESET good
    ```

13. Press "Start Ranking" button. Analysis process will be started. It can take a long time to process depending on encoder settings and hardware performance. You can see errors and warnings in the output log. Additionally encoder/decoder logs can be found in files "*d:\vicos\temp\codec_log.txt*" and "*d:\vicos\temp\codec_err.txt*".
14. Press key "View report…" to see results after analysis is finished.
15. Analysis results will be saved and used in later launches. If you want to recalculate results, press "Clear results" button at the main window.

6 Analysis Report Description

An example of main page of the report is depicted at figure below.



Pic. 10 The main page of analysis report

It consists of three parts:

- Overall Results diagram

- Analyzers Results diagram
- Table with Analyzers results

Several separate analyzers works during codec analysis. They generate some marks from 0 to 100 for each codec. Zero means the worst codec among tested, 100—the best one. These marks are shown in the table at the end of the page.

Total score column is separate analyzers marks averaging among all the analyzers. Averaging weights are fixed.

Diagrams simply visualize marks of separate analyzers and total marks.

Detailed information about analyzer's marks can be obtained by clicking on marks in table. Each mark is linked with detailed analyzer report.

6.1 Overall Analyzer Detailed Report

Pic. 11 En example of overall analyzer detailed report

Detailed report of overall analyzer contains table with relative PSNR and SSIM values for all tested sequences. Marks are relative to reference codec/preset, which is x264/default. For example, value 0.69 means that the test codec/preset requires 31% more bitrate to encode test sequence with the same quality. Last row of the table is averaged relative mark for all the sequences.

Analyzer's mark construction is described above in section "4.1 Overall Analyzer". Note, that this is raw analyzer mark, not fitted to segment [0, 100].

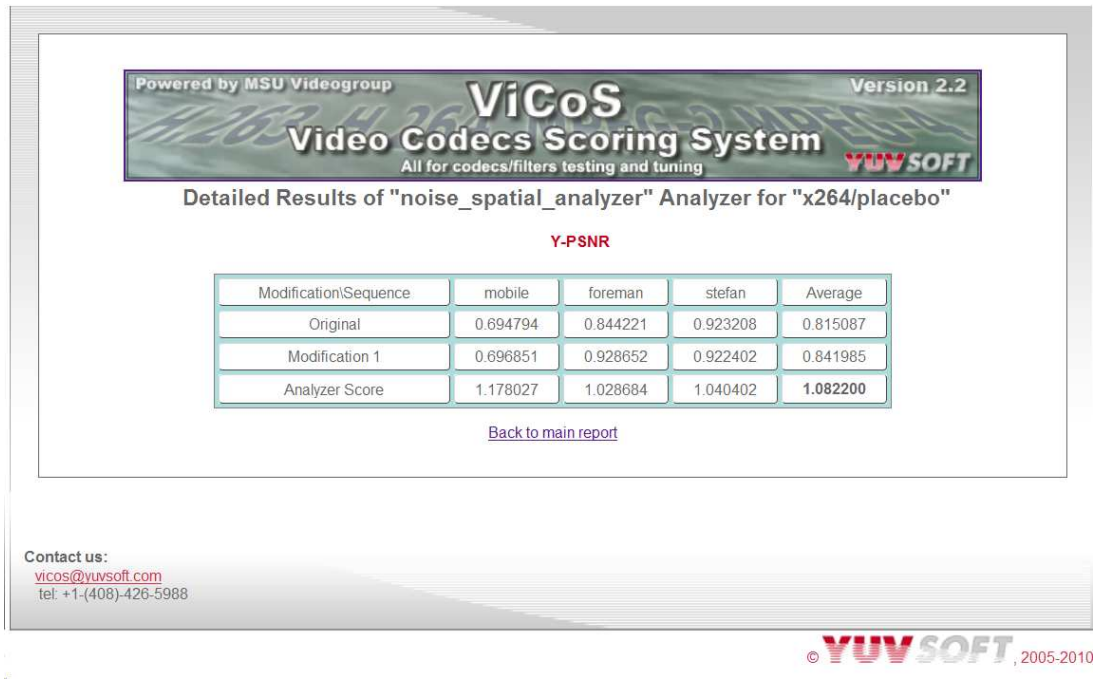
6.2 Synthetic Motion Analyzer Detailed Report

Pic. 12 En example of synthetic motion analyzer detailed report

Detailed report of synthetic motion analyzer contains table with relative PSNR values for different versions of synthetic motion sequence. The only sequence, used in this analyzer is “motion_1” sequence, which is synthetically generated sequence with moving rectangles. “Original” and “Modification 1” from the first row stand for sequence modifications with different strength of motion. Column “Average” always repeats column “motion_1”.

Analyzer’s score construction is described above in section “Motion Analyzer”. Note, that this is raw analyzer mark, not fitted to segment [0, 100].

6.3 Noisy Blocks Analyzer Detailed Report



Pic. 13 En example of spatial noise detailed analyzer report

Detailed report of spatial noise contains table with relative PSNR values for original versions of natural sequence and their modifications with spatial noise. Each column corresponds to single natural sequence (except the last one, which is averaged value for all sequences). Row “Original” contains relative marks for original sequences; row “Modification 1”—for marks of sequences with added noise.

Analyzer’s score construction is described above in section “Spatial Variable Noise Analyzer”. Note, that this is raw analyzer mark, not fitted to segment [0, 100].

7 Troubleshooting

Strange or “N/A” codec marks in report are indicator of some errors in most cases. There are several ways to understand reasons of those errors:

- First, analyze log of the system. Reading of warnings and errors can lead to bugs understanding.
- See detailed reports. Sometimes N/A in overall report can appear because of very pure quality of the encoder.
- Encoder and decoder output are not included to the system log, but stored in the following files: <*DataFolder*>*temp**codec_log.txt* and <*DataFolder*>*temp**codec_err.txt* (standard output and log streams). Analysis of these files is very fruitful if you are not sure in preset’s batches correctness.
- Encoded streams can be found in folder <*DataFolder*>\encoded_streams\. Note, that decoded files are not stored.
- If you are sure that tested codec works properly, but you still can’t include it to the system, write us an e-mail to vicos@yuvsoft.com. Add as much information about the codec as possible! We will try to help you!