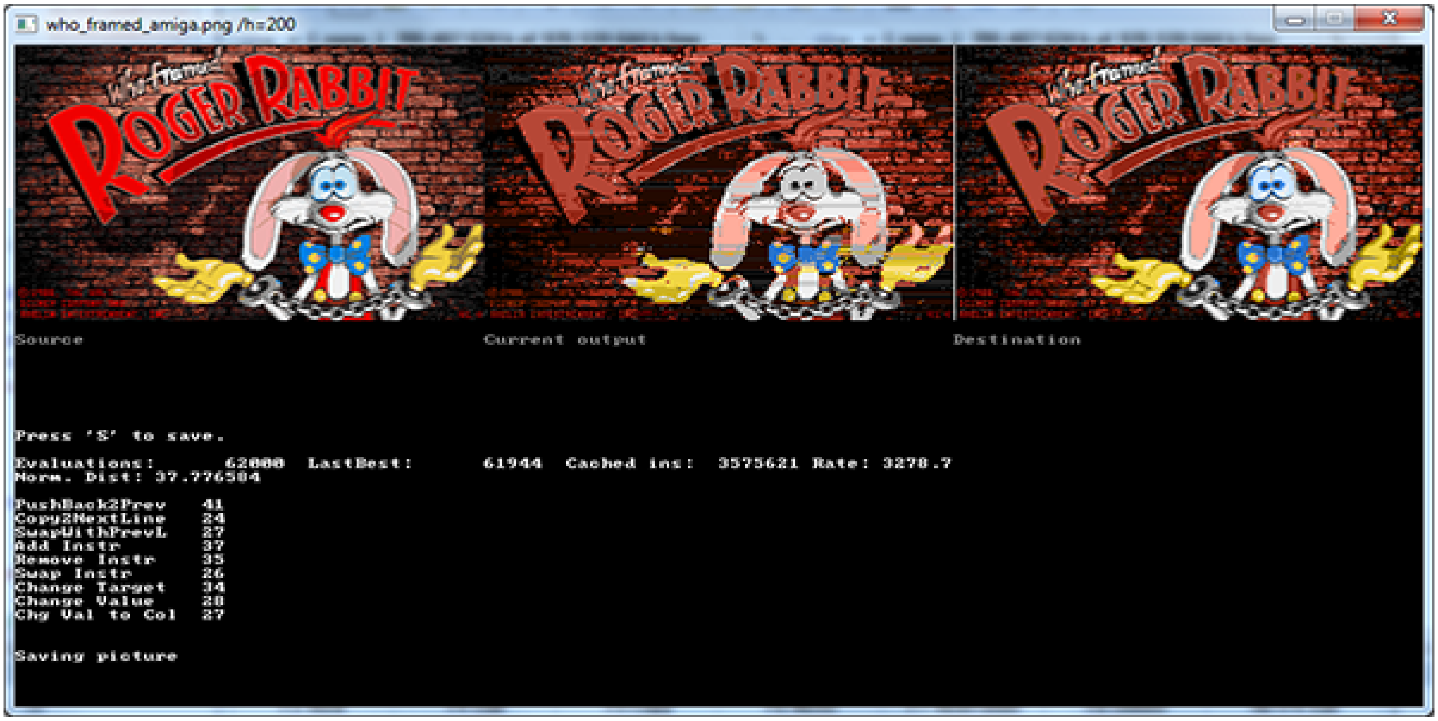
RastaConverter

by Jakub 'Ilmenit' Debski

About:

RastaConverter is a graphics converter from modern computers to old 8bit Atari computers.
The tool uses Allegro and FreeImage graphics libraries.
The conversion process is optimization of the Kernel Program. It uses most of the Atari graphics capabilities including sprites, midline color changes and sprite multiplication.
The converter uses Late Acceptance Hill Climbing algorithm.

Screenshot:



```
Press 'S' to save.

Evaluations:       62000  LastBest:       61944  Cached ins:  3575621 Rate: 3270.7
Norm. Dist: 37.776584

PushBack2Prev    41
Copy2NextLine    24
SwapWithPrevL    27
Add Instr        37
Remove Instr     35
Swap Instr       26
Change Target    34
Change Value     28
Chg Val to Col   27


Saving picture
```

Examples:

Atari executables for those and many other pictures can be downloaded here.

Usage:

```
RastaConverter.exe InputFile [options]
```

To save the output during the optimization process press 's' key.

Copy created output files to the Generator directory and run 'build.bat' to create executable file.

The application displays three images:

- The *left* one is the original picture resized to Atari width.
- The *right* one is the original picture mapped to chosen Atari palette. This is our goal to reach.
- The *central* one is preview of the output picture.

If the colors on the right image are wrong (f.e. too gray) then try some different palette file or remap original picture to chosen palette file using other tool f.e. Timanthes. Below on the screen are some statistics from the optimization process. With the current algorithm the longer you wait, the better results you get. For some pictures more than 1 million of evaluations is needed to get really good results.

Command-Line Options:

/i=Input File

File to process. All the formats of the FreeImage library supported, including JPG, PNG, GIF and BMP. Warning: it is not possible to load 16 colors pictures (4 bits per pixel). Increase the color depth first using other graphics program.

/h=Output Height

(Default: Proportional to picture size rescaled to 320xheight or 240.)

Generated file height. The speed of the optimization process depends on the height.

/continue

Continue previously stopped process. Currently it uses previously generated files (output*.*) therefore be careful not to overwrite them with other instance of RastaConverter.

/pal=Palette File

(Default: Palettes\laoo.act)

The output quality depends a lot on similarity between chosen Atari palette and the image palette. The default palette covers a large color range and is close to real Atari palette. You can also use shortcuts: `/pal=laoo` and the converter will try to load palette from '`Palettes\laoo.act`'

/dither=Dithering Type

[Options: chess, floyd, 2d, jarvis, simple, knoll]

- **'chess'** is Chessboard Dithering - typical 8bit dithering technique
- **'floyd'** is Floyd-Steinberg dithering - more "random" than chessboard dithering
- **'2d'** and 'simple' are between chess and floyd.
- **'jarvis'** dithering is more artistic
- **'knoll'** is similar to Adobe patented pattern dithering algorithm. Very slow with `/predistance=ciede`
- **'line'** dithering that tries to keep low number of color changes in line
- **'line2'** similar to 'line'

/dither_val=Dithering strength

(Default: 1, should be set from 0.0 to 3.0.)

Some dithering algorithms set to `/dither_val=2` may increase color saturation if the output picture is too gray.

/dither_rand=Dithering randomness

(Default: 0, should be set from 0.0 to 1.0.)

Current implementation of dithering algorithm produces output which may look too patterned. Set `/dither_rand` to 0.5 or 1.0 to make output more random.

/details=Details picture

To help the optimization algorithm you can set the details mask. The brighter the color on the mask is, the higher is distance of the color difference between input and output file on the masked region. This way you can make some objects like faces, eyes or other small objects more detailed.

/details_val=Details influence strength

(Default: 1)

/filter=Resize filter

(Default: box)
[Options: lanczos, box, bicubic, bilinear, bspline, catmullrom]

Filter used to resize the picture to the Atari width=160 color pixels (40 bytes).

/brightness=[-100,100]

(Default: 0)

Brightness color correction.

/contrast=[-100,100]

(Default: 0)

Contrast color correction.

/gamma=[0.0-8.0]

(Default: 1.0)

Gamma color correction.

/init=Initial optimization state

(Default: random)

The optimization search space is large and the final results depends a lot on the initial values. You may see that RastaConverter sometimes optimizes not those details of the picture that you want. To change the initial value you may use other states: random, empty, less, smart

'`/init=less`' option sets less colors in the initial state of the picture. This gives better result quick, but then picture is harder to optimize than with `/init=smart`

/o=Output File

(Default: 'output.png')

Output file name. Using a different name than the default, you have to change the file names used in the Generator.

/s=Number of solutions

(Default: 1)

You may set a different number of solutions used in the optimization process. With `/s=1` (default) the optimization algorithm is heuristic Hill Climbing. With more solutions the "Late Acceptance Hill Climbing" algorithm is used (www.cs.nott.ac.uk/~yxb/LAHC/) It works slower at the beginning, but after many (100M+) evaluations, when basic Hill Climbing almost stops, this algorithm still finds better solutions! The more solutions you use, the better final result can be. There is no memory impact so you can use even `/s=10000` or more.

/distance=Color distance function

(Default: yuv, other)
[Options: euclid, ciede, cie94]

- **euclid** - Use Euclidian RGB distance for colors instead of default YUV distance. This function is fast.
- **yuv** - produces usually better color mapping than Euclidian, but for some pictures RGB distance is also useful.
- **ciede** - uses the best visual color distance algorithm CIEDE2000, but can be very slow.
- **cie94** - something between slow ciede2000 and fast yuv

/predistance=Color distance function for preprocess

(Default: ciede)
[Options: euclid, yuv, cie94]

This option sets the preprocess color distance function to create destination picture. For the rest of the optimization the distance set by the `/distance` is used. The default option `/predistance=ciede` is VERY slow with Knoll dithering.

/max_evals=Maximum number of evaluations

RastaConverter will save the current solution and exit when this limit is reached.

/onoff=OnOff File

Text file that describes where to turn on and turn off usable registers. The format of each line in "OnOff file" is:

```
REGISTER OFF|ON FROM TO
```

REGISTER is one of the following registers: COLOR0, COLOR1, COLOR2, COLBAK, COLPM0, COLPM1, COLPM2, COLPM3, HPOSP0, HPOSP1, HPOSP2, HPOSP3

OFF turns off the usage of registers in lines, ON turns on. ON is default for all the registers in the whole picture.

FROM is line number in picture starting from 0

TO is line number in picture up to Picture Height-1

Every line in "OnOff file" is processed one after another.

Example:
To turn off usage of player 0 and player 1 in picture of height 240 pixels use the following OnOff file:

```
COLPM0 OFF 0 239
HPOSP0 OFF 0 239
COLPM1 OFF 0 239
HPOSP1 OFF 0 239
```

/preprocess

If this switch exists then RastaConverter generates and saves only destination picture.

/seed=seed for the random number generator

/save=number of solutions

(Default: 0)

Saves best solution after every 'n' evaluations. This is useful for a long-run optimizations, when the run can be stopped f.e. by power outage. If you run many instances of RastaConverter `/o` option may be useful.

Example:

```
RastaConverter.exe c64.png /threads=8 /s=1000
```