Source	Poznań University of Technology, Poznań, Poland
Status	Input document
Title	Slightly faster IVPSNR
Author	Jakub Stankowski, Adrian Dziembowski

Abstract

The document presents faster version of IVPSNR. The output of new version is exactly the same, as for IVPSNR v2.1.1, so they can be used interchangeably. Recommendations: * create IVPSNR 3.0 based on this proposal, * modify CTC in software tools section: v2.1.1 -> v2.1.1 or higher.

1 IV-PSNR v3.0 software changes

The goal of the work was to further decrease computational time (when compared to IVPSNR v2.1.1) without **any** change of the results. Source code is available on MPEG Git repository (dev tag).

1.1 CMake

- enabled INTERPROCEDURAL_OPTIMIZATION,
- assumed x86-64 Microarchitecture Feature Level >= x86-64-v2.

1.2 YUV file reader changes

- new implementation of picture I/O,
- reduced filesystem burden (avoid repetitive open-seek-read-close cycles).

1.3 Metrics calculation optimizations

- use of the interleaved picture layout for IVPSNR calculation:
  - improves memory locality,
  - most beneficial for high resolution (~4K) sequences,
- SIMD (SSE 4.1) implementation of most data processing functions.

1.4 Thread-level parallel processing

- dedicated thread pool instead of OpenMP directives (due to high OpenMP overhead),
- since v3.0 the OpenMP is no longer used.

1.5 Other improvements

- easy-to-use scrips for "one click" configure and build (*configure_and_build_IVPSNR.bat* and *configure_and_build_IVPSNR.sh*).

2 Compilation requirements

The IVPSNR v3.0 software uses following external components:

- "Formatting library for C++" (libfmt) – distributed under BSD licence and included in IVPSNR source package.

In order to build the software, the ISO C++17 conformant compiler is required.

3 Application parameters

3.1 Commandline parameters

Commandline parameters are not changed when compared to v2.1/v2.1.1.

3.2 Compile-time parameters

The IVPSNR v3.0 introduces two new compile-time parameters. Those parameters are defined in `CommonDef.h` file:

- **USE_INTERLEAVED_PIC** – Enables the usage of interleaved picture layout for IVPSNR calculation. (default = enabled),
- **USE_SIMD** – Enables the usage of SIMD (SSE 4.1) implementation of most data processing functions (but only if C++ compiler exposes presence of extensions by predefined macro). (default = enabled).

4 Results

4.1 Outputted quality

The results obtained by proposed version are exactly the same as for IVPSNR v2.1.1.

4.2 Processing time comparison

Calculated using 6-core CPU.

	Time [s]			
	v1.0	v2.0	v2.1	v3.0
SA	257.36	84.64	17.05	14.73
SB	127.48	42.01	8.63	7.34
SC	235.01	83.77	17.02	14.66
SD	66.88	22.30	4.62	3.92
SE	59.95	17.69	3.83	3.62
SJ	56.67	17.71	3.86	3.60
SL	56.53	17.73	3.87	3.60
Total	859.88	285.84	58.87	51.47

	Computation time reduction			
	v1.0-v2.0	v2.0-v2.1	**v2.1-v3.0**	cumulative
SA	67%	80%	**14%**	94.3%
SB	67%	79%	**15%**	94.2%
SC	64%	80%	**14%**	93.8%
SD	67%	79%	**15%**	94.1%
SE	70%	78%	**6%**	94.0%
SJ	69%	78%	**7%**	93.6%
SL	69%	78%	**7%**	93.6%
Total	67%	79%	**13%**	94.0%

	Relative speedup			
	v1.0-v2.0	v2.0-v2.1	**v2.1-v3.0**	cumulative
SA	3.04	4.96	**1.16**	17.47
SB	3.03	4.87	**1.18**	17.37
SC	2.81	4.92	**1.16**	16.03
SD	3.00	4.83	**1.18**	17.06
SE	3.39	4.61	**1.06**	16.56
SJ	3.20	4.59	**1.07**	15.74
SL	3.19	4.59	**1.07**	15.70
Total	3.01	4.86	**1.14**	16.71

5 Recommendations

We recommend to:
- create IVPSNR 3.0 based on this proposal,
- modify CTC in software tools section: v2.1.1 -> v2.1.1 or higher.

6 Acknowledgement