PERFORMANCE BENCHMARKING OF RVC BASED MULTIMEDIA SPECIFICATIONS

Junaid Jameel Ahmad[1,3], Shujun Li[2], Marco Mattavelli[3]

{Junaid.Ahmad, Marco.Mattavelli}@epfl.ch, Shujun.Li@surrey.ac.uk

[1] University of Konstanz, Germany, [2] University of Surrey, UK, [3] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Quick Questions and Answers

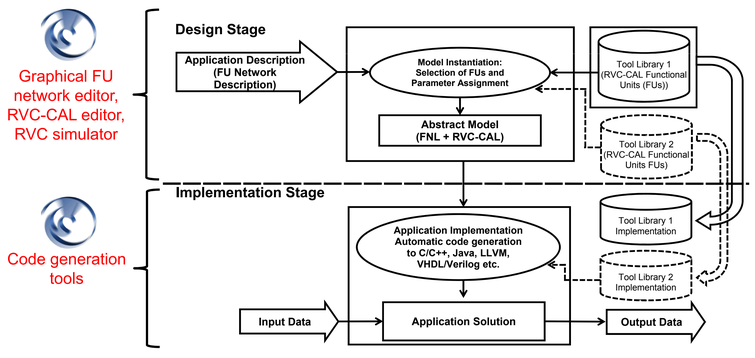
1. *What is RVC (Reconfigurable Video Coding)?*
   A dataflow programming framework standardized by ISO/IEC since 2009.
   - ISO/IEC 23001-4 (MPEG-B Part 4), 2009 (1st Edition), 2011 (2nd Edition) describes how to specify video codecs using three languages – RVC-CAL, FNL and RVC-BSDL.
   - ISO/IEC 23002-4 (MPEG-C Part 4), 2010 (1st Edition) describes a library of common tools used in MPEG video codecs.
2. *What is an RVC specification?*
   An implementation-agnostic description of a system specified by RVC.
3. *What have you done in this work?*
   We report performance benchmarking results of RVC-based C implementations of an H.264/AVC intra codec and a JPEG codec and four multimedia security systems working with them.
4. *How did you benchmark the performance of the RVC systems?*
   By presenting a side-by-side comparison of the RVC-based implementations against non-RVC reference implementations.
5. *What are the main results you observed?*
   - RVC implementations have a performance similar to non-RVC ones.
   - On a dual-core machine we observed a performance gain up to 173%.

1. How does RVC work?

Development Environment: eclipse + Open RVC-CAL Compiler

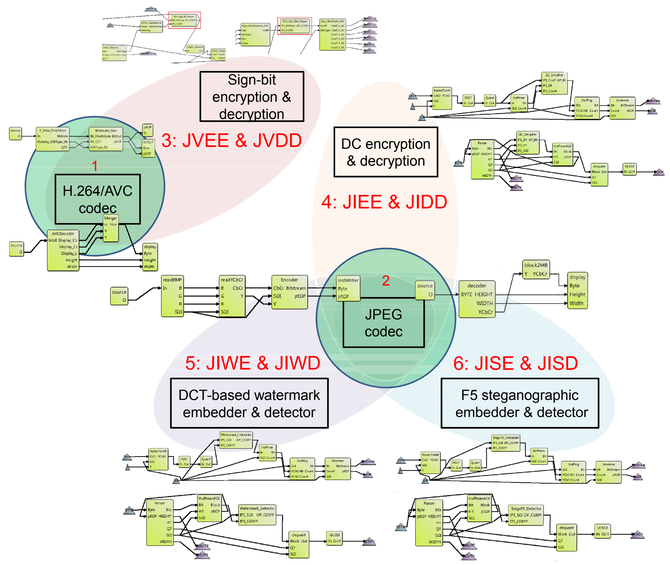
Graphical FU network editor, RVC-CAL editor, RVC simulator

Code generation tools



2. Benchmarked Systems

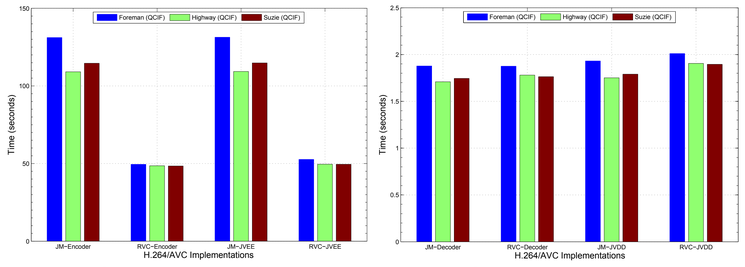
- **RVC-based C implementations**: specified manually in RVC by the RVC community and C code *automatically* generated from the RVC specifications by ORCC's CAL2C backend.
- **Non-RVC C implementations**: JM (for H.264/AVC) and IJG (for JPEG), C code manually written/optimized/maintained.



3. Experimental Results

3.1 Run-time performance on a single-core machine

H.264/AVC



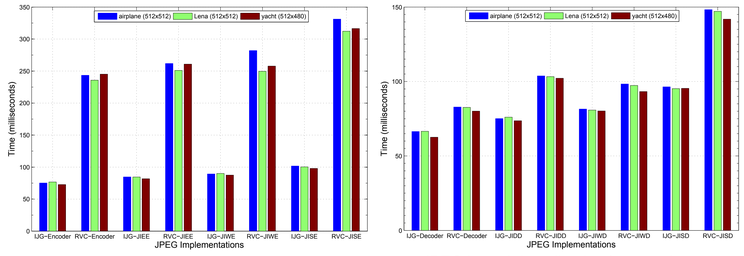
H.264/AVC based encoding implementations | H.264/AVC based decoding implementations

Test Video	JM-JVEE	JM-JVDD	RVC-JVEE	RVC-JVDD
Foreman	0.16%	2.77%	6.01%	6.73%
Highway	0.18%	2.41%	1.98%	6.55%
Suzie	0.17%	2.54%	2.25%	6.97%

Overheads of sign-bit encryption module

- RVC-based encoding implementations are (> 100%) faster.
- RVC-based decoding implementations are comparable to JM ones.
- RVC implementations suffer more encryption/decryption overhead.

JPEG



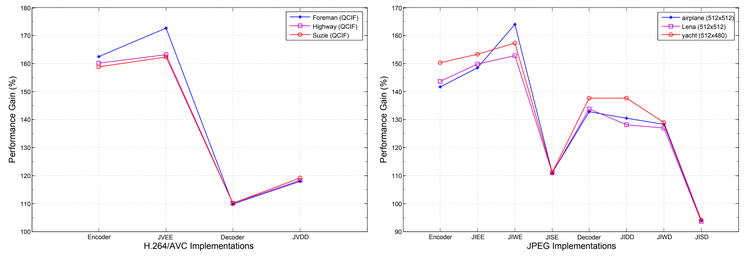
JPEG based encoding implementations | JPEG based decoding implementations

Test Image	JIEE	JIDD	JIWE	JIWD	JISE	JISD
RVC						
airplane	7.02%	20.13%	13.67%	15.73%	26.46%	44.11%
Lena	6.06%	20.00%	5.60%	15.13%	24.50%	43.85%
Pepper	6.00%	21.59%	4.91%	14.09%	22.52%	43.52%
IJG						
airplane	11.25%	11.55%	15.84%	18.50%	26.08%	31.12%
Lena	9.28%	12.50%	14.80%	17.66%	23.52%	30.11%
Pepper	11.13%	15.01%	16.85%	21.98%	27.77%	34.39%

Overheads of security modules

- RVC-based implementations are slower.
- JISE/JISD have the highest overhead due to image buffering.

3.2 Performance gain on a dual-core machine



Performance gain of H.264/AVC implementations:

- Encoding implementations: 159% – 173%.
- Decoding implementations: 110% – 119%.

Performance gain of JPEG implementations:

- Encoding implementations: 142% – 164%.
- Decoding implementations: 127% – 148%.

ICIP 2013 – 20th IEEE International Conference on Image Processing, Melbourne, Australia, September 15-18, 2013.