ISAMon: A Tool for Cross-Platform Performance Estimation and Clone Detection Using a Novel FFT-based Code Signature

Shubhankar S. Singh and Dr. Smruti R. Sarangi,

Indian Institute of Technology Delhi

Biography

- Shubhankar Suman Singh
- Ph.D. in CSE (2016 – present)
- IIT Delhi

- Dr. Smruti Ranjan Sarangi
- Associate Professor in CSE & EE
- IIT Delhi

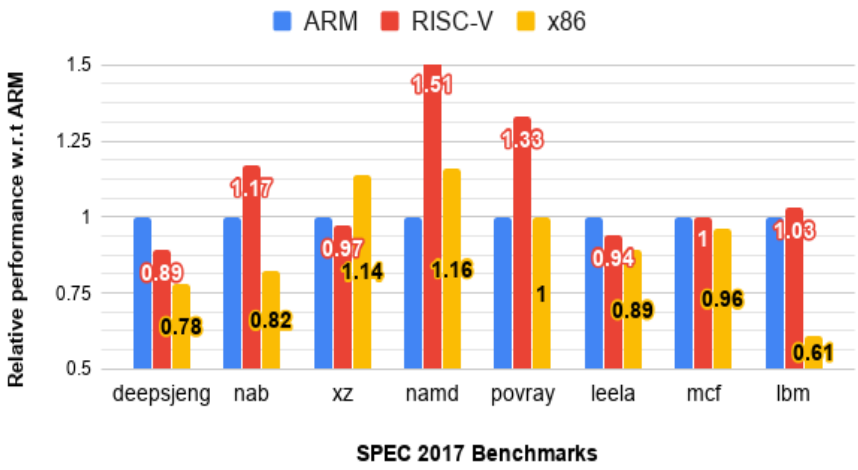
Overview

Many ISAs

- x86 (CISC)
- RISC-V, ARM (RISC)

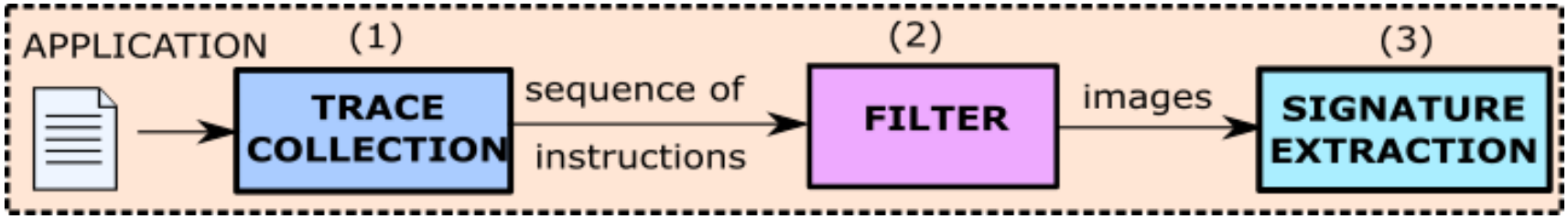
Why compare ISAs ?

- Performance
- Energy
- Create a multi-ISA architecture

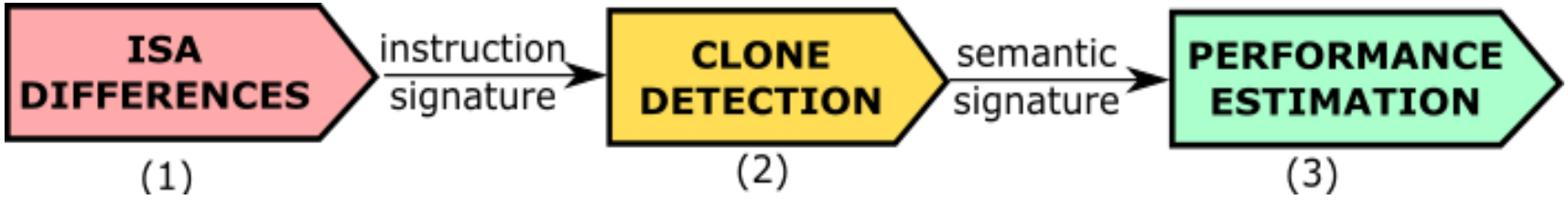


Performance difference: 0.6 to 1.5

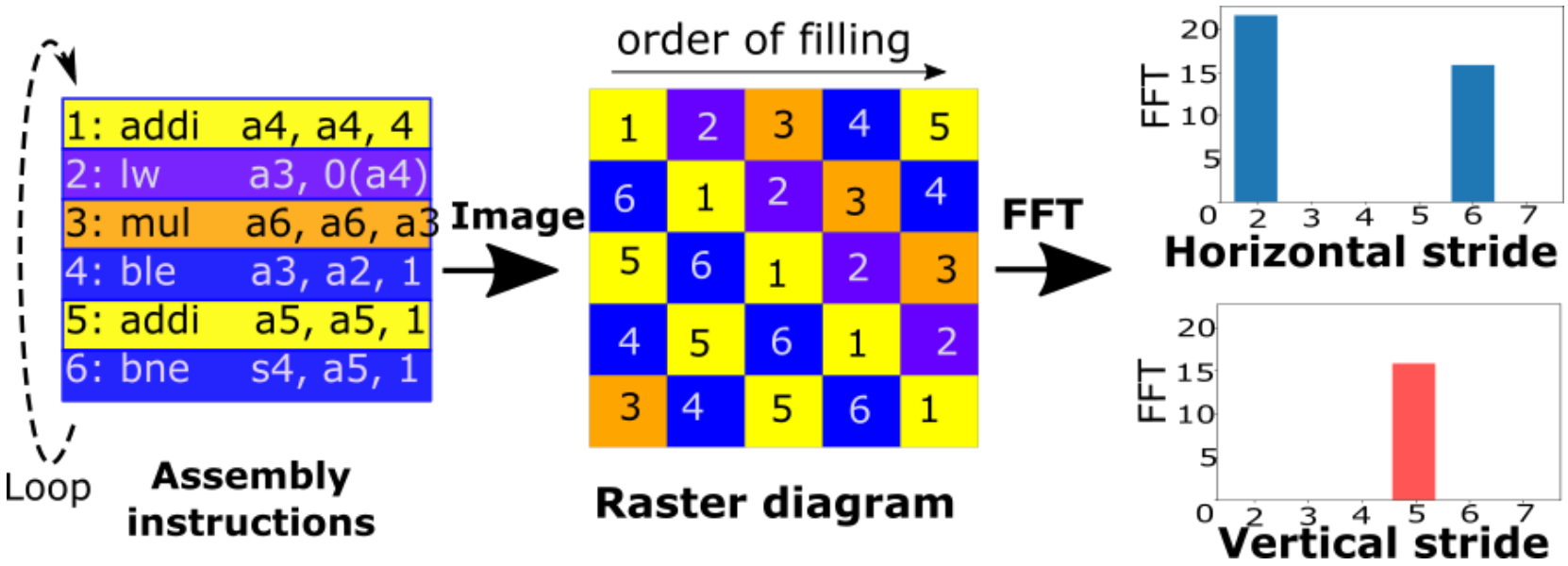
ISAMon Tool



- Application



Code Signature

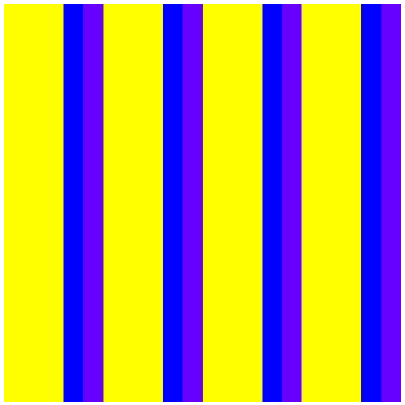


- Convert sequences of instructions into a raster diagram.
- Apply the Fourier transform on the raster image to obtain a horizontal and a vertical stride.

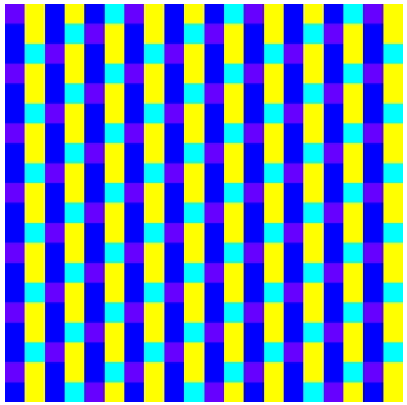
Code signature is a 5-tuple (H, V, C, P, I)

- H: Horizontal stride of the raster image (loop size)
- V: Vertical stride of the raster image (loop count)
- C: Color composition (histogram of number of instructions for each color)
- P: Phase type (most frequent instruction type)
- I: Instruction set architecture (ARM, RISC-V, x86)

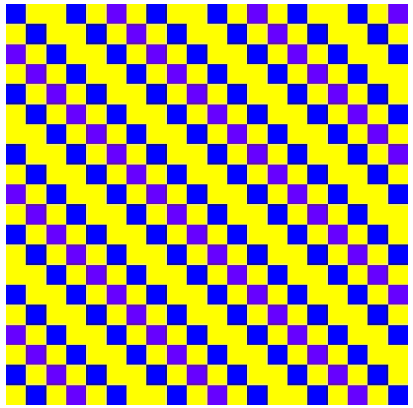
ISA Compare: *deepsjeng* (SPEC 17)



ARM=1



RISC-V=0.89



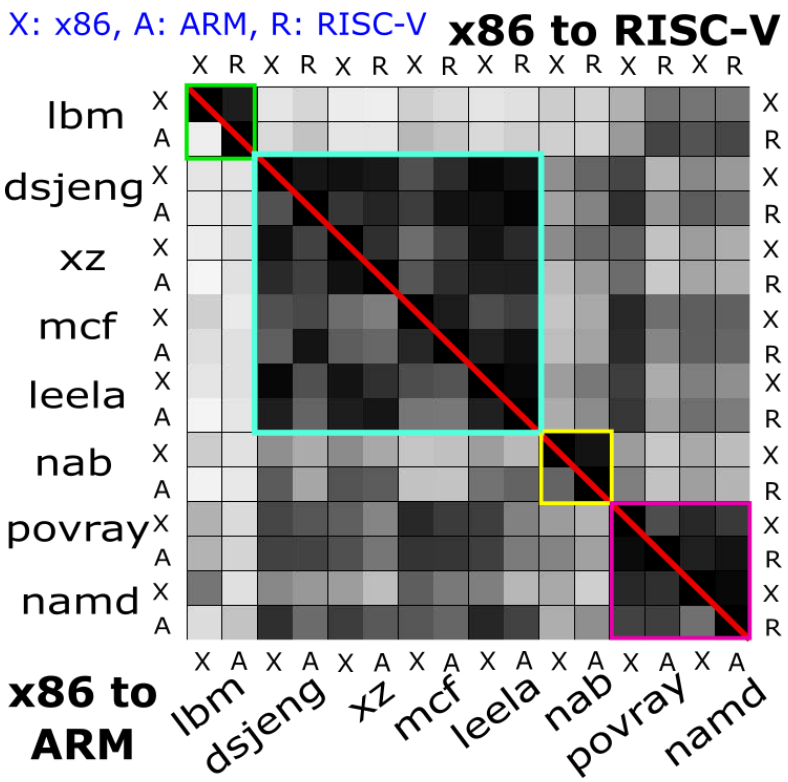
x86=0.78

Relative performance

Code signature	ARM	RISC-V	x86
Horizontal stride	5	6	7

Smaller loop size leads to a better performance

Clone Detection



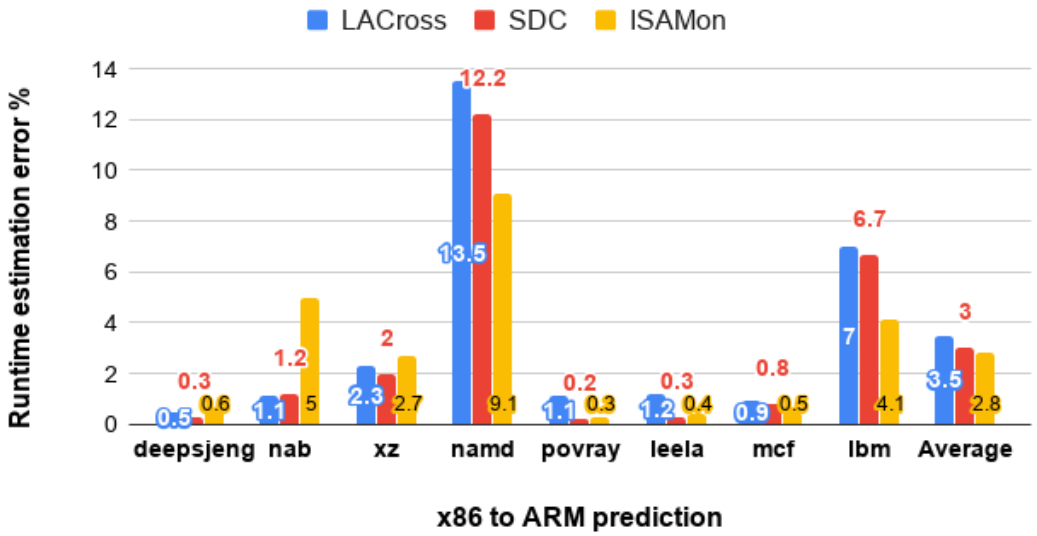
Confusion matrix

- Cross architecture clone detection for x86, ARM and RISC-V

Tools	Accuracy
ISAMon	84 %
kam1no [1]	46 %
BinDiff [2]	51 %

Performance Estimation

- Cross architecture performance estimation for x86, ARM and RISC-V
- **Algorithm**: piece-wise linear regression for estimating performance using code signature
- Compared against LACross and SDC



Tools	Error
ISAMon	2.8 %
LACross [3]	3.5 %
SDC [4]	3 %

Conclusion

- We proposed a novel FFT based code signature
- We proposed a visualization technique for a piece of code based on observed dynamic instructions.
- We used the code signature for comparing ISAs, detecting clones and estimating performance.
- Our algorithms are based on simulation techniques and hence we do not require the real hardware.
- The prediction phase of our algorithm is 5 times faster as compared to previous work.

References

[1] S. H. Ding, B. C. Fung, and P. Charland, “Asm2vec: Boosting static representation robustness for binary clone search against code obfuscation and compiler optimization,” in S&P, 2019.

[2] Bindiff version 5, https://www.zynamics.com

[3] X. Zheng, L. K. John, and A. Gerstlauer, “Accurate phase-level crossplatform power and performance estimation,” in DAC, 2016.

[4] X. Zheng, H. Vikalo, S. Song, L. John, and A. Gerstlauer, “Samplingbased binary-level cross-platform performance estimation,” in DATE’17.

Questions