Aggressive Data Transfer Reduction by loop fusion and re-computation for dedicated accelerators
Peemen, M.C.J.; Mesman, B.; Corporaal, H.

Published: 01/01/2015

Document Version
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):
Aggressive Data Transfer Reduction by Loop Fusion and Re-Computation for Dedicated Accelerators
Maurice Peemen, Bart Mesman, and Henk Corporaal

So-called "Smart Companion Devices" are Not Intelligent Yet!
Many smart companion devices, but all without intelligence. Deep Neural Nets and Machine Learning brings Superhuman recognition accuracy [1, 2] on complex tasks to wearables.

Huge data transfer requirements prevent efficient implementation of such intelligent functionalities with acceptable battery lifetime.

HLS Based Accelerators for Energy Efficiency
Demanding apps require highly customized accelerators to achieve challenging throughput and energy constraints. Energy is often dominated by data movement and operator energy is neglectable. State-of-the-art uses loop tiling and interchange [3, 4] to optimize data reuse in local accelerator buffers.

Looking Beyond Tiling: Fusion and Re-Computation
Tiling or classical fusion with row buffers does not work well, limited reuse or large buffer requirements. Introducing redundant workload could help out.

Design Space Exploration
Manual exploration intractable, therefore automate the search through solution space. We obtain optimized schedules regarding locality and buffer size.

The Fruits of Our Approach
Apply the suggested transformations to the filters and synthesize with HSL, mapping results to Virtex 6 FPGA at 150 MHz.

- Fast Design Space Exploration (seconds)
- Up to 2x communication reduction w.r.t. [4]
- Up to 8x better throughput

References