Review of Dissertation Thesis

Title: Lifting Scheme Cores for Wavelet Transform

Author: David Bařina, Faculty of Information Technology, Brno University of Technology

I was given to review the above stated Ph.D. thesis dated from December 2015. The thesis has 118 numbered pages and consists of eight chapters (Discrete wavelet transform, Computation schedules, Lifting vectorization, Lifting core, Multi-Dimensional cores, Evaluation, and Conclusion).

In the thesis, the author focuses on effectively computing the wavelet transform by making use of so called lifting schemes. The topic can be regarded as useful since the need for computing the wavelet transform arises quite frequently, e.g., in JPEG 2000 compression. At the same time, the topic by no means is easy. Many authors have already presented their works in this area.

The structure of the thesis can be regarded as more or less classical. In Chapter 2, the basics of the wavelet transform and lifting schemes are summarized for the convenience of the reader. Chapter 3 is devoted to the problems of practical computation. The chapter also covers the two-dimensional wavelet transform. Various architectures are mentioned, including SIMD extensions (SSE, AVX), GPUs, and FPGAs. Works of many authors are cited in this chapter. The chapter may also be regarded as a state-of-the-art chapter. In Chapter 4, various ways of how the lifting vectorization can be done are discussed. The main contribution of the thesis (as the author claims) is creating a unit based on the lifting scheme technique. The unit can be characterized by the possibility of reorganizing the operations and by a variety of allowed processing orders. The unit is called the core by the author. The cores can also run in parallel. The presented core approach can be extended to multiple dimensions (Chapter 6). The multidimensional cores can also run in parallel. Chapter 7 is devoted to evaluation of the performance. Firstly, the performance without SIMD vectorization was examined. The results were compared with the results presented by Rade Kutil in the paper “A Single-Loop Approach to SIMD Parallelization of 2-D Wavelet Lifting”. From Table 7.1, a certain improvement is noticeable, which holds especially for the AMD processors. Several other evaluations are carried out in Chapter 7, e.g. the influence of the order of processing and the influence of the core size are studied. Certain results of processing on GPUs and FPGAs are presented too.

Generally, the work is well organized and well written. The problem is important, which is mainly due to its presence in image compression. The appropriate results of the author are presented. The contribution has been presented in several conference papers and in one journal paper (Journal of Real-Time Image Processing). Also, the results of the implementations on GPUs and FPGAs are given. From this point of view, I regard the author's scope as relatively broad. If I can judge, the work is written without typos. From what I have written above, it follows that my overall evaluation of the work is positive. In the following part of this review, I present only a minor remarks.

With respect to the results of comparison with the algorithm proposed by Rade Kutil (Table 7.1), it would be nice to have a transparent list of differences or similarities between both the algorithms. I confess that I understood the differences better from the paper in JRTIP than from the thesis itself. The results should probably deserve some discussion, e.g., what is the source of so different speedup factors for the Intel and AMD processors. The author claims that the experiments were focused on x86 architecture; it would be nice to state how it was exactly achieved.

I believe that the results stated in Table 7.10 would also deserve at least a short discussion explaining how the differences in performance should be understood. Is it a matter of how the approaches are sophisticated or is it a matter of how the technical platforms are good? Are the

results comparable if the age of the results is so different? More details could be given on how the FPGA implementation was exactly realized. The reader would probably be interested.

The comments in a similar spirit I also have regarding the GPU implementation. The corresponding subchapter claims that the core method outperforms the state-of-the-art method. It seems to me that it is a pity that a more comprehensive and more convincing comparison was not presented if the author expended that effort and created the implementation. By the way, more details on how the GPU implementation was done would also be interesting.

More details could also be given about the implementation (practical realization) of cores. If the author claims that the advantage is a certain flexibility of cores, it would be nice to show whether and how this flexibility is achieved in practice (it would also stress the importance of the approach). Finally, I would also welcome a compact, clear, and rigorous (more or less) definition of the term core, which could also be accompanied with a brief list of similarities and differences with other approaches if possible.

Summary: The thesis focuses on an important and demanding topic. The author has proven his ability of research work. He proposed a solution based on so-called cores, the solution was evaluated experimentally, the experiments have shown that the solution is useful. The ideas presented in the thesis were appropriately presented in conference papers and in a journal paper. If I had comments and remarks, they were of minor importance. On the basis of all this, *I do recommend the thesis for the defense*.

Ostrava, April 19, 2016

doc. Dr. Ing. Eduard Sojka
FEI, VŠB-TU Ostrava