

SpringerBriefs in Computer Science

Series Editors

Stan Zdonik
Peng Ning
Shashi Shekhar
Jonathan Katz
Xindong Wu
Lakhmi C Jain
David Padua
Xuemin Shen
Borko Furht
VS Subrahmanian
Martial Hebert
Katsuchi Ikeuchi
Bruno Siciliano

For further volumes:

<http://www.springer.com/series/10028>

Ben Juurlink • Mauricio Alvarez-Mesa
Chi Ching Chi • Arnaldo Azevedo
Cor Meenderinck • Alex Ramirez

Scalable Parallel Programming Applied to H.264/AVC Decoding

Ben Juurlink
Technische Universität Berlin
Berlin, Germany
juurlink@cs.tu-berlin.de

Mauricio Alvarez-Mesa
Technische Universität Berlin
Fraunhofer HHI.
Berlin, Germany
alvarez@ac.upc.edu

Chi Ching Chi
Technische Universität Berlin
Berlin, Germany

Arnaldo Azevedo
Delft University of Technology
Delft, The Netherlands

Cor Meenderinck
IntelliMagic. Leiden
The Netherlands

Alex Ramirez
Universitat Politècnica de Catalunya
Barcelona Supercomputing Center
Barcelona, Spain

ISSN 2191-5768

ISBN 978-1-4614-2229-7

DOI 10.1007/978-1-4614-2230-3

Springer New York Heidelberg Dordrecht London

ISSN 2191-5776 (electronic)

ISBN 978-1-4614-2230-3 (eBook)

Library of Congress Control Number: 2012938716

© The Author(s) 2012

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

What is scalability?

Title of an article authored by Mark Hill that appeared in SIGARCH Comput. Arch. News, 18(4), Dec. 1990.

Preface

Welcome to this book! In it we share our experiences in developing highly efficient and scalable parallel implementations of H.264/AVC video decoding, a state-of-the-art video coding standard. We hope to convince you, the reader, that scalable parallel programming is an art and that substantial progress is still needed to make it feasible for non-expert programmers. We also present a parallel-application-design-process and hope that this design process will make the development of parallel applications easier.

When we were invited by Springer to write a SpringerBrief book because our article “Parallel Scalability of Video Decoders” was among the top-downloaded articles from the Journal of Signal Processing Systems, we accepted the invitation but did not want to simply extend the original article. Instead, we decided to present all or at least most of the work we did on this very exciting topic in order to be able to present a complete picture. This was easier said than done. For one, the work was done over a time span of several years and in several years the state-of-the-art computing systems change dramatically. Because of this, the computing systems used to evaluate the presented implementations may differ in each chapter. Please forgive us for these inconsistencies. If we had had more time and money, we would have done it differently.

This book is targeted at graduate students, teachers in higher education, and professionals who would like to understand what it means to parallelize a real application. While there are many textbooks on parallel programming and parallel algorithms, for understandable reasons of space, very few discuss real applications. It is also targeted at video coding experts who know a lot about video coding but who would like to know how it could be parallelized and which features of modern multi-/many-core architectures need to be exploited in order to develop efficient implementations. When reading this book they will probably smile because we use some of their terms wrongly or in a different context than they usually do. Well, we will smile back at them when they confuse an SMP with a cc-NUMA or vice versa.

This book may be used in several ways. For example, it may be used as a supplement to a parallel programming, parallel computer architecture, or parallel algorithms course. One of the authors uses this material to give two lectures about “The

Art of Parallel Programming - H.264 Decoding as a Case Study” at the end of a course on multicore systems (slides including exercises are available on request). This was also the title we originally had in mind for this book, but Springer thought that this title was too long. We accepted this advice as they are, after all, in the publishing business.

While reviewing this book one of the authors mentioned that we are too negative, that this book might scare away people from parallel programming. This is certainly not our intention, and we have revised the book to make it more positive. On the contrary, by presenting a parallel-application-design-process we hope to interest more people in the art of parallel programming.

This book was mainly written by the first three authors. The other three authors, however, contributed significantly to the articles on which this book is based, and therefore they are rightfully mentioned as co-authors. We especially would like to thank our families (Claudia, Lukas, Leon, Claudia, Luna, Ozana, Alex, Marieke, Alicia, Helena, Martí) for their love and support. This book had to be written partially in our spare times, which should be devoted to them. We would also like to thank Senj Temple, the first author’s Canadian sister-in-law, for proofreading several parts of this book. None of the authors is a native English speaker and her feedback was really helpful. Also thanks to Biao Wang for his help with the encoding of some of the videos used in this book and for providing a nice IDCT example.

Berlin,
March 2012

*Ben Juurlink, Mauricio Alvarez-Mesa,
Chi Ching Chi*

Acknowledgements

This work described in this book was supported in part by the European Commission in the context of the SARC integrated project, grant agreement no. 27648 (FP6), the ENCORE project, grant agreement no. 248647 (FP7), and the European Network of Excellence on High-Performance Embedded Architecture and Compilation (HiPEAC).

Contents

- 1 Introduction 1**
 - References 3
- 2 Understanding the Application: An Overview of the H.264 Standard . 5**
 - 2.1 Introduction 5
 - 2.2 High-level Overview 6
 - 2.3 Elements of a Video Sequence 8
 - 2.4 Frame Types 9
 - 2.5 H.264 Coding Tools 9
 - 2.5.1 Entropy Coding 9
 - 2.5.2 Integer Transform 10
 - 2.5.3 Quantization 11
 - 2.5.4 Inter-Prediction 11
 - 2.5.5 Intra-Prediction 12
 - 2.5.6 Deblocking filter 13
 - 2.5.7 Comparison With Previous Standards 13
 - 2.6 Profiles and Levels 13
 - 2.7 Conclusions 15
 - References 15
- 3 Discovering the Parallelism: Task-level Parallelism in H.264 Decoding 17**
 - 3.1 Introduction 17
 - 3.2 Function-level Decomposition 18
 - 3.3 Data-level Decomposition 19
 - 3.3.1 Frame-level Parallelism 19
 - 3.3.2 Slice-level Parallelism 20
 - 3.3.3 Macroblock-level Parallelism 21
 - 3.3.4 Other Data-level Decompositions 31
 - 3.4 Conclusions 32
 - References 33

4	Exploiting Parallelism: the 2D-Wave	35
4.1	Introduction	35
4.2	Cell Architecture Overview	36
4.3	Task Pool Implementation	37
4.4	Ring-Line Implementation	41
4.5	Experimental Evaluation	46
4.5.1	Performance and Scalability	46
4.5.2	Profiling Analysis	48
4.6	Conclusions	51
	References	52
5	Extracting More Parallelism: the 3D-Wave	53
5.1	Introduction	53
5.2	Dynamic 3D-Wave Algorithm	54
5.3	2D-Wave Implementation on a Shared-Memory System	55
5.4	Dynamic 3D-Wave Implementation	58
5.5	Experimental Evaluation	60
5.5.1	Experimental Setup	60
5.5.2	Experimental Results	62
5.6	Conclusions	65
	References	66
6	Addressing the Bottleneck: Parallel Entropy Decoding	67
6.1	Introduction	67
6.2	Profiling and Amdahl	68
6.3	Parallelizing CABAC Entropy Decoding	71
6.3.1	High-Level Overview of CABAC	72
6.3.2	Frame-level Parallelization of CABAC	73
6.4	Experimental Evaluation	75
6.4.1	Experimental Setup	75
6.4.2	Experimental Results	76
6.5	Conclusions	78
	References	78
7	Putting It All Together: A Fully Parallel and Efficient H.264 Decoder	81
7.1	Introduction	81
7.2	Pipelining H.264 Decoding	82
7.3	Parallel Entropy Decoding	84
7.4	Parallel Macroblock Reconstruction	85
7.5	Dynamic Load Balancing using Unified Decoding Threads	86
7.6	Experimental Evaluation	91
7.7	Conclusions	95
	References	96
8	Conclusions	97
	References	101

Acronyms

AVC	Advanced Video Coding
CABAC	Context Adaptive Binary Arithmetic Coding
CAVLC	Context Adaptive Variable Length Coding
CRF	Constant Rate Factor
DCT	Discrete Cosine Transform
DLP	Data-level Parallelism
DMA	Direct Memory Access
DPB	Decoded Picture Buffer
EDT	Entropy Decoding Thread
EIB	Element Interconnect Bus
GOP	Group of Pictures
HEVC	High Efficiency Video Coding
MB	Macroblock
MPEG	Moving Pictures Expert Group
MRT	Macroblock Reconstruction Thread
MV	Motion Vector
MVP	Motion Vector Prediction
NUMA	Non-Uniform Memory Architecture
ORL	Overlapping Ring-Line
PPE	Power Processing Element
QP	Quantization Parameter
RL	Ring-Line
SIMD	Single Instruction Multiple Data
SPE	Synergistic Processing Elements
SPMD	Single Program Multiple Data
SSB	Slice Syntax Buffer
TLP	Thread-level Parallelism
TP	Task Pool
UDT	Unified Decoding Thread
VCEG	Video Coding Experts Group