Low Power Techniques for Video Compression

Valentin Muresan[†], Noel O'Connor[†], Noel Murphy[†], Sean Marlow[†] and Stephen McGrath[*]

[†] *Center for Digital Video Processing*
Dublin City University
IRELAND

[§] *Multimedia Business Unit*
Parthus plc
IRELAND

E-mail: [†]Valentin.Muresan@dcu.ie [*]Stephen.McGrath@parthus.com

Abstract — **This paper gives an overview of low-power techniques proposed in the literature for mobile multimedia and Internet applications. Exploitable aspects are discussed in the behavior of different video compression tools. These power-efficient solutions are then classified by synthesis domain and level of abstraction. As this paper is meant to be a starting point for further research in the area, a low-power hardware& software co-design methodology is outlined in the end as a possible scenario for video-codec-on-a-chip implementations on future mobile multimedia platforms.**

Keywords — **mobile multimedia, video compression, low power, hardware acceleration**

I Introduction

This paper will focus on the increasingly important set of embedded applications to run on portable systems in the areas of digital communications and multimedia consumer electronics, e.g., cellular phones, personal digital assistants and multimedia terminals. These complex systems rely on "power hungry" algorithms for wireless communications, video compression and decompression [1], image processing, etc. Figure 1 depicts what are expected to be the main functional peripherals of the next generation mobile multimedia platforms. We see from this that hardware accelerating solutions need to be designed to support future real-time mobile applications for MPEG-4 coding/decoding, GPS localization, and Bluetooth, UMTS, GSM/GPRS wireless communications. The portability of mobile platforms makes energy consumption a particularly critical design concern as it reduces battery life. Moreover, high power dissipation leads to more expensive packaging and decreases reliability.

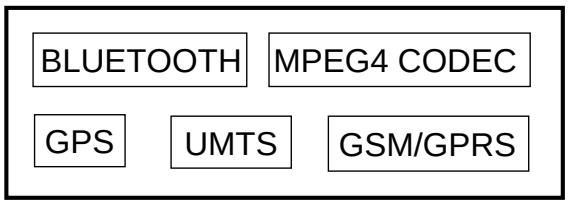

Fig. 1: Next Generation Mobile Platforms' Peripherals

The potential for multimedia applications on mobile platforms is constrained by low bandwidth wireless connections, low computational power, low memory capacity, and short-life battery problems. The first problem is ameliorated by efficient video compression standards as in MPEG-4. The computational requirements of MPEG-4, however, exacerbate the remaining three problems, which are themselves strongly inter-related. For example, the software implementations of video compression tools are time consuming and could not cope with the needs for real-time results required by applications like video-conferencing. The real-time applications require high-throughput hardware accelerators to speed up computationally demanding tools. The hardware solutions designed solely for high-throughput are a lot faster, but usually take more power than their software counterparts. Therefore, the short-life battery problem is aggravated when the high-throughput hardware acceleration is compulsory and, hence, the short-life battery problem becomes the biggest problem for mobile multimedia.

The next section describes how features of MPEG-4 can be the basis for power/performance efficiency of future mobile multimedia platforms. Section III enumerates power-efficient hardware solutions already proposed or potentially applicable for video compression acceleration. Section IV describes how power saving can be achieved at a

high-level by exploiting the behavioral peculiarity of each video compression tool. Finally a possible System-on-a-Chip scenario for mobile multimedia is sketched based on a top-down hardware/software (HW/SW) co-design methodology.

II Shape Adaptive Video Compression

MPEG-4 was proposed as a standard to meet the scalability/flexibility requirements of different mobile multimedia platforms. Therefore, it provides a large set of tools that can be selectively applied. The tools are divided into many overlapping sets, called profiles. The MPEG-4 Simple Profile (SP) provides, for example, the most commonly used tools and is closely related to H263. A highly efficient compression standard like MPEG-4 pays the price for its object-orienting advantages by employing computationally expensive algorithms for motion estimation (ME), discrete cosine transform (DCT), discrete wavelet transform (DWT), CAE (Content-based Arithmetic Encoding) binary-shape coding, variable length coding (VLC), quantisation (Q).

The basic idea of previous video compression standards (MPEG-1, MPEG-2) is to employ motion estimation and DCT tools in order to eliminate temporal and spatial redundancy. MPEG-4 introduces new tools in order to boost compression efficiency and to provide new object-based functionality. The introduction of object shape along with its texture and motion features is one of the main steps forward. The shape of a video object is represented by a pixel-resolution binary alpha-plane, which is coded by CAE binary shape coding. The coded shape is used as the basis for compression in a number of ways, for example, the polygon matching technique for motion estimation/compensation. Polygon matching assumes that only pixels within the shape of the video object are considered in the matching criterion used in the motion estimation search strategy. We can use this information to disable the useless computation invested in the unnecessary compression of outside-shape parts. Shape-adaptive versions of the DCT are another example of compression tools where unnecessary computation can be avoided to save power.

III Low Power Hardware Architectures

Several architectural solutions have already been proposed for implementing video compression tools in hardware [2] including dedicated Application Specific Integrated Circuit (ASIC) solutions, Digital Signal Processing (DSP) architectures, reconfigurable Single Instruction Multiple Data (SIMD) based hardware, Field Programmable Gate Array (FPGA) implementations and circuit-level technological domain techniques. Unfortunately, only a few of these are power-conscious. FPGA technologies cannot yet meet mobile device's power, miniaturization, and speed requirements. Also, technological domain techniques are usually employed close to the foundry and, therefore, are beyond the scope of this overview. However, voltage-scaling and dynamic clock frequency are two circuit-level technological domain techniques known to be power-efficient. Even though video compression tools can be implemented by means of regular hardware (SIMD and systolic arrays), the video content and its associated processing and compression are highly non-uniform in both space and time. Therefore, the efficiency of such high-throughput solutions decreases dramatically in the case of video compression. High-throughput solutions are also very power hungry. Unless high-throughput is a necessity for the highest rates or it is already available for other mobile multimedia functions (GSM/GPRS/UMTS,GPS), regular hardware solutions are not too likely to be appropriate for power-efficient video compression. Consequently, dedicated (ASIC) DSP solutions are the most promising approach in order to achieve the level of performance and power consumption appropriate to mobile multimedia applications.

Power efficiency of computational hardware can be dealt with in the behavioral or structural domains. Examples of power optimization techniques carried out on the behavioral specifications so that the video compression tools become power-conscious are: power-aware scalability, and motion estimation by adaptive block-matching and power-conscious search algorithms. These are usually achieved at algorithmic level, by ordering/reducing the basic operations involved in the compression tools so that the switching activity is minimized, or decreasing the levels of computation (e.g., getting rid of the enhancement layer video data processing) when the power consumption levels go over a given limit. Such techniques sometimes achieve lower power consumptions levels at the expense of poorer quality video. The scalability of the compression tools is controlled at system level and involves only software-based decisions. These follow the paradigm of power/distortion-optimized and power/rate-optimized compression strategies. On the other hand, the adaptive block-matching and power-conscious search algorithms for motion estimation involve HW/SW co-design solutions (see

section V).

In the structural domain, power optimization decisions can be taken at system, register transfer (RT), or logic levels. At system level the low power techniques consist of the reconfiguration of datapath, memory, system bus and control units. In reconfigurable architectures, the control unit usually has a system configuration part that is in charge of the on-line or off-line reconfiguration of the system components: datapath, memory, system bus. At RT and logic levels, pipeline re-structuring and word truncation are typical low-power techniques.

However, there are also low-power techniques that necessitate a tight HW/SW co-design methodology in order to achieve efficient designs. For example, MPEG-4's object shape and texture processing means a requirement for hardware flexibility enhancement in order to follow the arbitrarily-changing size and shape of the object being compressed. The arbitrarily shaped object mechanism has behavioral connotations that can be accelerated only by a highly reconfigurable power-efficient hardware architecture flexible enough to follow an object' shape run-time characteristics.

Power-efficient hardware solutions potentially applicable to the video compression acceleration issue are briefly described next.

a) Low Power Reconfigurable Architectures

A variety of solutions have been proposed as configurable and programmable architectures for video compression. They can be classified in four main categories [3]: circuit-level technological reconfigurability, gate-level reconfigurability (FPGAs), logic level reconfigurability of the functional modules, parametrical reconfigurability of the functional modules (memory, bus, or datapath bit-sizes), and reconfigurable by programmability. The programmability reconfigurable architectures proposed for video compression are based on general programmable processors with or without DSP/multimedia extensions. They are normally power-inefficient and are not discussed here because they are beyond the scope of this paper. The parametrical reconfigurability technique of the functional modules assumes usually a run time reallocation of the hardware resources so that the parts which are not involve in the processing are disabled or shut-down. These techniques are usually mixed up with the logic level reconfigurability techniques.

a).1 Voltage-Scaling Techniques

In [4] voltage-scaling technological approaches are summarized as examples of circuit-level technological techniques that can be employed in DSP domain: firstly, globally scaling the supply voltage along with the threshold voltage, secondly, a dual-V_{dd} approach in which the reduced V_{dd} is selectively applied to non-critical paths, and thirdly, a variable supply voltage approach, where the V_{dd} is controlled on-chip adaptively. This approach is one of the most efficient, but it can be achieved only at circuit-level in the technological domain.

a).2 Low-Power Programmable Architectures

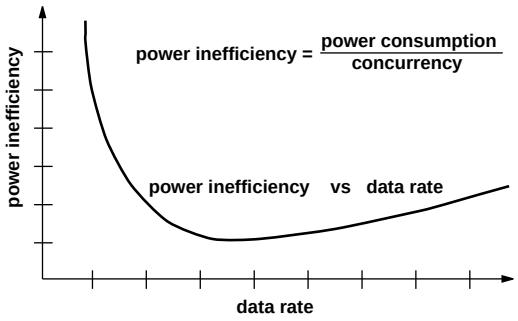

Fig. 2: Power vs Data Rate

FPGAs allow parallelism, pipelining, local memory and both functional and data dedication. However, FPGAs suffer from disadvantages for mobile applications, such as the difficulty of miniaturization, higher power consumption and their slowness at sequential computations. FPGAs are not designed to support high-speed dynamic reconfiguration as they exhibit a delay overhead given by the reconfiguration mechanism. Other similar circuit-level (technological) reconfigurability solutions have also been proposed recently in [5] for dynamic interconnect architectures. These solutions have not reached their maturity because of the long delays exhibited by the programmable interconnections logic. FPGAs are not power efficient due to their high level of programmability and their lack of support for memory-intensive computation [6], even though low-power FPGAs have been recently proposed [7] for DSP.

a).3 Low-Power by Datapath Flexibility

Several parallel architectures have already been applied to the datapath structure of video compres-

sion: SIMD arrays [2] and pipelining are amongst the most popular. The SIMD architectures are used as hardware accelerators for high-throughput DSP applications. They are very efficient for applications where a constantly high processing level is required. Even though video compression can be implemented by means of regular DSP hardware, the video content and its associated processing and compression are highly non-uniform in both space and time [6]. This means that much of the time, the SIMD hardware is consuming power but not carrying out useful processing, as the average video rates are significantly lower than the maximal ones.

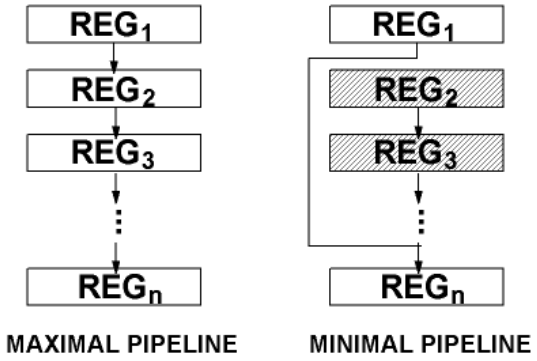

Fig. 3: Dynamic Pipelining

The highly pipelined VLSI architectures designed for high data rates of video compression are excessively power consumptive at low rates (see figure 2). Figure 2 depicts the power inefficiency curve drawn against a fluctuating video data rate for a certain level of parallelism or concurrency (e.g., pipelining) [8]. Similar curves can be drawn for any level of parallelism (e.g., number of pipeline stages). A power efficient parallel structure would be one able to dynamically reconfigure so that the minimum possible level of power is consumed for any given video data rate. Pipeline throughput scalability is the fashionable solution in this case and it translates in hardware terms to architecture flexibility. For example, in [8] a reconfigurable datapath solution is proposed as the one depicted in figure 3. Here the maximal pipeline can be employed to achieve high throughput when the video data rates are high. In the case of lower video data rates, the number of pipeline stages can be reduced so that the power is consumed efficiently. This can be achieved by disabling and bypassing an appropriate number of dissipative pipeline stages according to the data rate. For the lowest video data rate the minimal pipeline structure can be employed.

Fundamentally, the above dynamic pipeline technique saves power by eliminating pipeline stages when the high processing-per-cycle rate is not justified. A simpler way to achieve the same results would be, for example, to run the full pipeline for 50% of the time and then shut down the pipeline for the rest of the processing, rather than configure a minimal pipeline with 50% of the stages disabled. This approach is architecturally simpler, but delivers power-unbalanced results. That is, in the above example, the pipelined architecture reaches a maximum power level for the first 50% of the computation and then the power level drops virtually to a zero power level.

Other logic-level low complexity and power techniques are the word-length shortening techniques used to truncate the pixel-value's bit-length when a high level correlation is exhibited in the input video data. These techniques are known to save computation complexity and indirectly power consumption, but sometimes they also degrade the compression rate because they lack the SAD estimation precision.

a).4 Low-Power On-Chip Memory

In general, the memory sub-architecture consumes a significant amount of power because of two sources of power loss: the frequency of memory access causes dynamic power loss, while leakage current also contributes to power loss. Organizing the memory so that an access activates only parts of it, helps at limiting dynamic memory power loss. Memory banking, currently used in some low-power designs, splits the memory into banks and activates only the bank presently in use. It relies on the exploitation of video-content spatial locality, which can be increased by studying and optimizing the video content reference pattern.

To avoid the leakage power loss, memory bank shut-down procedures could be employed on memory parts that are unused for long time. Other power-efficient techniques deal with the optimization of memory (bank) size and addressing hardware. In video compression, the dynamic power loss can also be reduced either by reducing the redundant access to video data or by reordering and grouping the independent operations of the compression tools so that the number of accesses to the same video-data element is reduced.

a).5 Low-Power Local Bus Architectures

Video compression tools are memory intensive. Therefore, local memory architectures are used to avoid system bus conflicts, lighten the system bus management, and speed-up the system level bus

transfers. These low power on-chip memory techniques are also meant to eliminate power-inefficient system-bus transfers. In the literature, buses have also proved to be a significant source of power loss, especially in the case of wide (32-64 lines) inter-chip buses, where each line requires substantial drivers. One approach employed to limit the switching on these lines is to integrate data compression techniques (e.g., Gray code for address lines or transmitting the difference between successive address values for address lines as well) with the bus controllers to eliminate the switching activity on the bus. This way the data compression/decompression is executed on the fly and reduces the power loss levels.

IV Low Power Behavioral Optimization

At a high level, the hardware/software architectures can be tailored to achieve low power consumption levels by synthesizing dynamically parameterized algorithms [6, 9]. These architectures are able to adapt at various rates of video information and its associated compression computations.

Other behaviorally power-optimized approaches suitable for video compression tools are the functional reconfigurable algorithms and architectures, which have the potential to reduce power consumption by adjusting their dimensions. For example, for the case of motion estimation tool, a dynamically resized memory can implement a flexible search area for motion vectors [10]). Then SAD calculation cancellation techniques can be employed again for the motion estimation search strategy and block-matching algorithms to cease the computation when the power consumption levels go over a certain limit.

Power savings can be achieved in the behavioral domain by employing the shape-adaptiveness feature of MPEG-4 tools described in section II. Polygon matching for motion estimation and shape-adaptive DCT/IDCT are the mostly known and they assume reducing the computational and, indirectly, the power consumption levels to the needs of compressing shape, motion and texture information for each arbitrarily-shaped object in video's scene.

V Low Power Mobile Multimedia Design Methodology

The complexity of video compression entails a System on a Chip (SOC) design methodology in order to embrace most of its functionality. Mobile multimedia applications bring yet another level of power vs performance constraints. Starting from each MPEG tool specification, investigations have to be made and an dedicated architectural view formulated for each of them based on their behavioral peculiarities. High-level hardware/software co-design decisions need to be made in order to decide the ratio between the hardware/software architectural solutions to be employed further down in the synthesis/implementation. Then the models have to be validated and tested to determine if their functionality meets the specifications. From this level, silicon products vendors can take over and bring the optimized SOC to silicon level.

VI Conclusions

Handheld manufacturers have already started to eye the digital multimedia applications as a huge source for value-added products to be sold on the future 3G wireless market. Therefore, they will slowly shift their focus into the video compression world to find the best price/power/performance balanced HW/SW implementations. This paper tackles this balance from the power consumption perspective and enumerates the low-power techniques known in the mobile multimedia literature.

References

[1] O'Connor, N. E. and Murphy, N. A. and Marlow S.: **Image and Video Compression - Module Notes** - School of Electronics, Dublin City University, 2000-2001.

[2] Muresan, V.: **Hardware Accelerating Solutions for Mobile Device Platforms** - Internal Technical Report, Video Media Processing Lab, Dublin City University, April 2001.

[3] Rabaey, J.: **Reconfigurable Processing: The Solution to Low-Power Programmble DSP**, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, April 1997.

[4] Usami, K. and Igarashi, M. and Ishikawa, T. and Kanazawa, M. and Takahashi, M. and Hamada, M. and Arakida, H. and Terazawa, T. and Kuroda, T.: **Design Methodology of Ultra Low-Power MPEG-4 Codec Core Exploiting Voltage Scaling Techniques**, Proceedings of the 35^{th} IEEE Design Automation Conference, June 1998, pp.483-488.

[5] Zhang, H. and Wan, M. and George, V. and Rabaey, J.: **Interconnect Architecture Exploration for Low-Power Reconfigurable Single-Chip DSPs**, Proceedings of the IEEE

Workshop on VLSI Signal Processing, Napa California, October 1992, pp. 166-174.

[6] Burleson, W. and Tessier, R. and Goeckel, D. and Swaminathan, S. and Jain, P. and Euh, J. and Venkatraman, S. and Thyagarajan, V.: **Dynamicall Parameterized algorithms and Architectures to Exploit Signal Variations for Improved Performance and Reduced Power**, Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), USA, May 2001.

[7] George, V. and Zhang, H. and Rabaey, J.: **The Design of a Low Energy FPGA**, ISLPED '99 - Proceedings of the 1999 International Symposium on Low Power Electronic Design, 1999, pp. 188-193.

[8] Kim, S. and Ziesler, C. H. and Papaefthymiou, M. C.: **A Reconfigurable Pipelined IDCT for Low-Energy Video Processing**, Proceedings of the 14^{th} IEEE International ASIC/SOC Conference, USA, Sep 2001.

[9] Wan, M. and Zhang, H. and George, V. and Benes, M. and Abnous, A. and Prabhu, V. and Rabaey, J.: **Design Methodology of a Low-Energy Reconfigurable Single-Chip DSP System**, Journal of VLSI Signal Processing, 2000.

[10] Park, S. R. and Burleson, W.: **Reconfiguration for Power Saving in Real-Time Motion Estimation**, Proceedings of ICASSP, 1997.