hevcDTM: Application-Driven Dynamic Thermal Management for High Efficiency Video Coding

[1,2]Daniel Palomino, [1]Muhammad Shafique, [1]Hussam Amrouch, [2]Altamiro Susin, [1]Jörg Henkel
[1]Chair for Embedded Systems (CES), Karlsruhe Institute of Technology (KIT), Germany
[2]Informatics Institute, PPGC, Federal University of Rio Grande do Sul (UFRGS), Brazil
dmvpalomino@inf.ufrgs.br, altamiro.susin@ufrgs.br, {muhammad.shafique, amrouch, henkel}@kit.edu

***Abstract* — This paper presents an application-driven algorithm for Dynamic Thermal Management (DTM) for the High Efficiency Video Coding (HEVC). For efficient design of such a DTM policy, we perform an offline thermal analysis of an HEVC encoder and demonstrate the impact of different video sequences and different coding configurations on the processor temperature. Our thermal analysis is leveraged to develop an efficient application-driven DTM policy that performs temperature-aware coding along with an application-driven control of DTM knobs (e.g., frequency scaling) in order to meet the temperature constraints while still providing high video quality (i.e. PSNR loss < 0.01dB). For accurate thermal analysis and evaluation, we deploy an infrared camera-based thermal measurement setup that, on the contrary to state-of-the-art setups, does not require adding any extra layer on top of the measured chip, thus allowing the camera to accurately capture the infrared emissions from the die.**

Keywords—Dynamic Thermal Management, DTM, HEVC, Application-Specific Optimization, Thermal Analysis, IR Camera.

I. Introduction and Related Work

Video encoding/decoding applications have become an integral part of mobile devices, servers, and PCs. The JCT-VC standardization committee has recently developed the next-generation *High Efficiency Video Coding* (HEVC) standard [1][2]. It provides 2x higher compression compared to the AVC/H.264 by incorporating an advanced set of novel coding tools like *Coded Tree Unit* (CTU) structure with large-sized blocks [3], numerous (Intra and Inter) prediction modes with an intensive mode decision search [4], and improved in-loop deblocking filters. However, these tools significantly increase the computational complexity, power, and temperature; see [6] for an extensive analysis of HEVC. Compared to H.264, the complexity of the block matching process with inter prediction mode decision in HEVC is up to ~2.2x [5].

Such high complexity can be well complemented by the advances in the semiconductor technology where continuous technology scaling enables developing modern multimedia computing platforms [7]. However, integrating billions of transistors in a small area has resulted in high power densities and consequently increased on-chip temperatures [8][9]. This, in turn, leads to elevated cooling costs and degraded reliability/lifetime [8][9][10]. *Therefore, temperature reduction is one of the primary design objectives in HEVC-based embedded multimedia systems to maintain a reliable operation during lifetime*.

In order to keep the temperature under safe operational limits (i.e. below a specified threshold [1]) or to lower the overall peak/average chip temperatures, Dynamic Thermal Management (DTM) policies are employed. These policies use task migration [11], frequency/voltage scaling [12]-[14], clock gating [15], or combination of these as control knobs [13]. Traditional DTM policies (such as [11]-[15], etc.) typically monitor the temperature and predict the workload to perform task migration or frequency/voltage scaling at run-time. However, these state-of-the-art DTM do not exploit the application-specific properties of video coding and may incur significant performance or quality of service (QoS) degradation (e.g., loss in video quality in terms of PSNR or frame rate, loss in bit rate, etc.) [13][18]. *In case of video encoding/decoding, the algorithmic and video content knowledge can be characterized and exploited to obtain more efficient DTM*.

State-of-the-art in DTM for video coding mainly target old standards like MPEG-2 and H.264, thus may not be efficiently applied to next-generation HEVC due to its novel coding tools like CTU and high complexity. The works in [16] and [17] perform spatial and temporal quality degradation using frame drops for MPEG2 decoding to meet the temperature constraints. These techniques primarily rely on frame drops for lowering the temperature which results in significant QoS loss that may not be acceptable in many scenarios. Moreover, they mainly target video decoding and do not account for encoding, which is >10x more complex than decoding [19] and consequently more challenging for DTM. In order to cope with the performance degradation effects of different DTM policies, the work in [18] selects a video quality degradation mode using encoder parameters. However, it does not investigate and address the impact of different video properties on the generated temperature.

In Summary: To address the aforementioned challenges there is a prominent need for an application-driven DTM policy that efficiently control the on-chip temperatures during the HEVC encoding while still providing a high video quality. Developing such a policy necessitates analyzing the HEVC thermal behavior using an accurate thermal setup.

A. Our Novel Contributions and Concept Overview

1) **hevcDTM – An Application-Driven Dynamic Thermal Management Policy for HEVC (Section III):** It performs a temperature-aware coding configuration selection along with an application-driven control of DTM knobs (like frequency scaling) in order to manage temperatures with minimum penalties in terms of bit rate as well as PSNR. This policy leverages the temperature impact of HEVC coding tools and video content properties.

2) **HEVC Thermal Analysis (Section II):** For designing an application-driven DTM policy, we perform an extensive thermal analysis of the HEVC standard and study the thermal behavior of different coding configurations and video sequences.

To the best of authors' knowledge, we are the *first to perform thermal analysis and management of the next-generation HEVC*.

[1] Typically threshold temperature as specified by the chip vendor.

978-3-9815370-2-4/DATE14/©2014 EDAA

II. THERMAL ANALYSIS OF THE HEVC ENCODER

Before moving to our HEVC thermal analysis, we present an overview of our in-house thermal measurement setup.

A. Peltier-Based IR Thermal Measurement Setup

To circumvent the information deficiency on the thermal distribution across modern processor chips and to accurately analyze their thermal behavior when executing complex real-world applications such as the HEVC, real-time thermal imaging of a processor using IR-cameras is essentially demanded [20][21][22]. It facilitates researchers and designers to develop, optimize, and evaluate efficient DTM policies. However, this requires safely removing the cooling system (e.g., fan, metal heat sink, and packaging) of the processor chip as it is an infrared opaque in order to expose its die for measuring the emitted thermal radiations. Consequently, to keep the processor temperature in a safe range without affecting the thermal imaging, alternate IR-transparent cooling mechanisms need to be used.

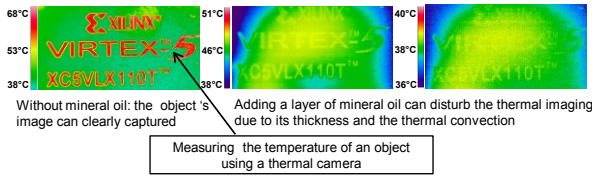


Fig. 1: The impact of mineral oil on thermal imaging; a special oil designed for the IR spectrum (similar to [20][21]) is used to demonstrate its impact when the thermal radiations emitted from a hot object are measured by an IR-camera.

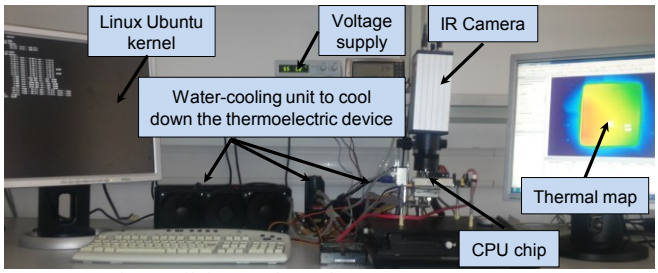


Fig. 2: Oil-free IR-thermal measurement setup [26].

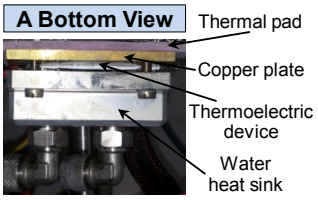


Fig. 3: Bottom-view of the chip showing the cooling mechanism [26].

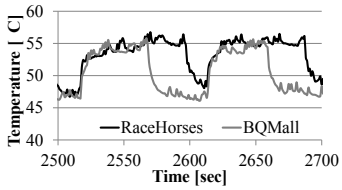


Fig. 4: Temperature cycles for two sequences

State-of-the-art works deploy thermal setups [20]-[22] introduce an additional transparent layer of mineral oil between the IR-camera and the die-under-measurement for cooling. This can disturb the thermal imaging due to its thickness and thermal convection (see Fig. 1). Therefore, we deploy an advanced oil-free measurement setup [26] that provides researchers more clear thermal imaging; see Fig. 2 and Fig. 3. It allows the processor to work within operational conditions and does not require adding any new layer between the die-under-measurement and the IR-camera lens. It employs a thermoelectric device that continuously cools down the chip *from the bottom side* (i.e. right below where the chip is located on the PCB) in order to maintain a safe operation. When the electrical current flows through the thermoelectric device, it generates a temperature variation between both device sides making one side cold and the second one hot due to the *Peltier* effect. Removing the released heat from the hot side results in reducing more temperature of the cold side. By tuning the voltage fed into the thermoelectric device, a wide range of applied chip cooling can be obtained to support different kinds of processors requirements.

In our experiments, we study an Intel Atom 45nm dual-core processor operating at a maximum frequency of 1.8 GHz. The on-chip temperatures are directly measured using a DIAS pyroview 380L compact IR-thermal camera capable of precisely capturing temperatures with an accuracy of ±1 °C and a spatial resolution of 50 µm per pixel [23]. The real-time thermal images were taken and sent to a PC (at a frame rate of 50Hz) that analyzes them to build the corresponding thermal maps of the measured processor over time.

B. Thermal Analysis of the HEVC Encoder

Now, we present our thermal analysis of HEVC encoding [24] for different video sequences, which provides hints for designing an application-driven DTM. Video sequence properties (like motion intensity, texture intensity, etc.) can directly affect the encoding process behavior and the computational effort needed to encode one sequence. Consequently, these properties will affect the thermal behavior for encoding videos.

Thermal Analysis for Different Video Sequences considering Core Idling: Fig. 4 shows the core temperature over time for two test video sequences: "*RaceHorses*" with high-motion (dark line) and "*BQMall*" with low-motion (gray line). For each frame same arrival time is assumed. Therefore, if one video frame is encoded earlier, the core is put in the *idle* state to reduce the temperature. Fig. 4 shows that for the high-motion "*RaceHorses*" sequence, the temperature stays high for most of the time due to less idle period, while for the "*BQMall*" sequence the temperature goes down earlier and stays low more due to a longer time. The average temperature of encoding the low-motion sequence is 2.5 ºC lower than encoding the high-motion one. This shows that reducing the workload directly influences the cooling period of a core. Hence, *the workload of a high-motion sequence can be curtailed to increase the cooling-down period*.

Thermal Analysis for Different Video Sequences considering Frequency Scaling: Besides core idling, an alternate method to reduce the average temperature is executing the workload at a low frequency while stretching the execution time so that the encoding finishes near the next frame arrival time; otherwise the core is put to the idle mode. Fig. 5 shows the core temperature over time for the same test video sequences when reducing the operating frequency by 25% of the maximum frequency (1.8 GHz to 1.35 GHz) for the low-motion sequence[2]. This results in a decrease in the peak/maximum core temperature from 56.4 ºC to 53.9 ºC. Fig. 5 also shows the thermal map[3] at two interesting points in time before going to the idle state, that the maximum temperature of the low-motion sequence is lower than that for the high-motion sequence.

Thermal Analysis for Different Video Sequences considering Parallelism in HEVC using Multiple Tiles: Another challenging scenario is considering tighter deadlines. In

[2] Voltage scaling is not available on our Atom board. However our approach and measurement setup is equally applicable to processors with DVS support.

[3] The thermal maps are mirrored with the die floorplan since this is how the infrared camera shows the images.

such a case the low-motion sequence may finish in time, but the high-motion sequence needs more performance which can be achieved by using the HEVC's tile-level parallelism, i.e. partitioning the frame into two parts and executing two threads on two cores. Fig. 6 shows the temperature over time for two sequences with a tighter deadline. The "*RaceHorses*" sequence is parallelized on two cores using two tiles to achieve high throughput so that both sequences finish their execution at the same time. If this is before the next frame's arrival time, the cores are put to the idle state (see sudden temperature drop). Using 2 cores for the "*RaceHorses*" sequence leads to 5 ºC higher temperature compared to the "*BQMall*" sequence. The temperature state of the chip at the end of frame encoding is shown using the thermal maps.

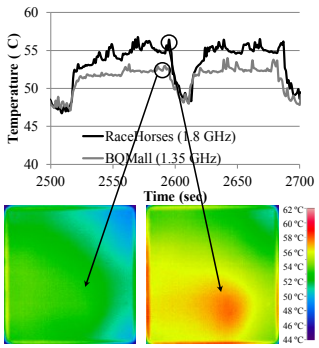


Fig. 5: Temperature analysis using FS for low motion sequence.

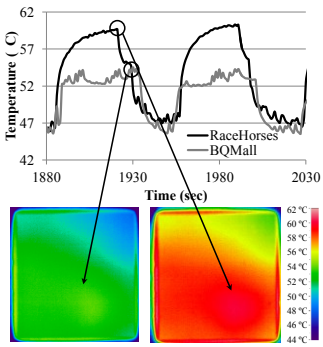


Fig. 6: Temperature analysis for high motion sequence with 2 cores.

The above experimental analysis demonstrates that video properties (especially the motion content) directly influence the generated temperature for HEVC encoding and potential of applying DTM mechanisms (e.g. core idling or frequency scaling). *Therefore, we aim at addressing the key challenge of leveraging the video content properties to enable application-level temperature management during the HEVC encoding.*

III. HEVCDTM: OUR APPLICATION-DRIVEN DTM POLICY FOR THE HEVC ENCODER

Fig. 7 shows our system overview with the HEVC encoder, our application-driven DTM algorithm, and the Atom processor with the temperature sensors. The main goal of our DTM algorithm is to leverage the video properties, encoder parameters, and other control knobs (like frequency scaling and task migration) to keep the current temperature below a threshold temperature and minimize the quality degradation. While the HEVC encoder is processing a sequence, our DTM algorithm monitors the current temperature through the available temperature sensors. In case the temperature is approaching to the pre-defined thermal threshold, it adapts the encoder parameters (like quantization parameter and CU size) to lower the temperature. Furthermore, our DTM algorithm extracts motion and texture properties of the video sequences and leverage them to further adapt the encoder parameters.

Fig. 7 also illustrates a high level flow of our proposed application-driven DTM algorithm. The algorithm performs the temperature control at two levels of the encoding process using motion intensity and texture properties: (1) long time system configuration (frame by frame); and (2) short time system configuration (CU by CU). The *long time system configuration* is based on the motion intensity variation between frames. The first step of the algorithm is to extract the motion intensity related to the target frame and depending upon the motion intensity classification, frequency scaling can be applied in case of medium and low motion sequences. Then, the algorithm chooses an appropriate set of parameters also based on motion intensity to start the frame encoding process. For the motion intensity extraction we deploy the method proposed in [25] using the following equations (1)-(3), where the thresholds to classify the motion intensity are obtained using probabilistic density function. We basically use the motion vector sizes from previous frames as input to the motion intensity calculation.

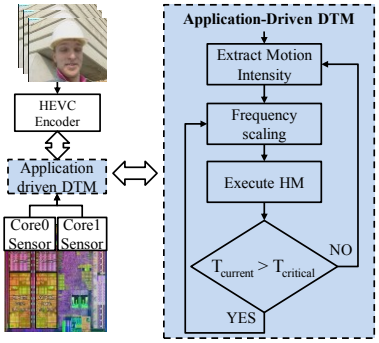


Fig. 7: System overview and our application-driven DTM algorithm.

$$\forall n \in M \qquad D_n = \sqrt{vx_n{}^2 + vy_n{}^2} \tag{1}$$

$$D = \sum_{\forall n \in M} D_n / size(M) \tag{2}$$

$$\Psi = \begin{cases} Low\ Motion\ (LM) & if(D \le Th_{D1}) \\ Medium\ Motion\ (MM) & if(Th_{D1} < D \le Th_{D2}) \\ High\ Motion\ (HM) & if(D > Th_{D2}) \end{cases} \tag{3}$$

After that, the *short time system configuration* is triggered. For each CU being encoded, the algorithm extracts its texture, and based on this spatial content property, the algorithm tune the encoder parameters targeting temperature, bit rate and video quality. In our algorithm the CU texture extraction (Eq. 5) is performed using the variance (Eq. 4) of the luminance samples (ρ_i in Eq. 4) within a CU, as proposed in [25].

While these long and short system configurations are being performed, the DTM algorithm also checks the system temperature. In case of thermal emergencies, the frequency scaling is used to accelerate cooling down of the system.

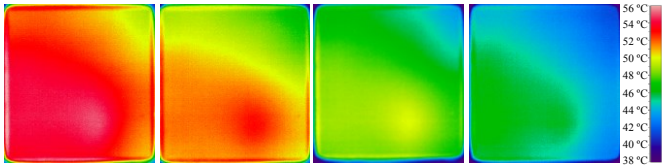
$$v_{CU} = \frac{1}{W_{CU} \times H_{CU}} \sum_{i=1}^{W_{CU} \times H_{CU}} (\rho_i - \rho_{avg})^2 \tag{4}$$

$$\Gamma = \begin{cases} Low\ Texture\ (LT) & if(v_{CU} \le Th_{v1}) \\ Medium\ Texture\ (MT) & if(Th_{v1} < v_{CU} \le Th_{v2}) \\ High\ Texture\ (HT) & if(v_{CU} > Th_{v2}) \end{cases} \tag{5}$$

IV. EXPERIMENTAL RESULTS

For evaluating our DTM algorithm, we used a set of five different sequences (*RaceHorses, Keiba, PartyScene, BasketballDrill* and *BQMall*) encoding with the HM 11.0 software using the same setup board as discussed in Section II.A. We evaluate our DTM policy for three threshold temperatures, namely 54ºC, 50ºC, and 46ºC showing how our proposed DTM algorithm adapts and study its effects on the PSNR and bit rate of the encoded video.

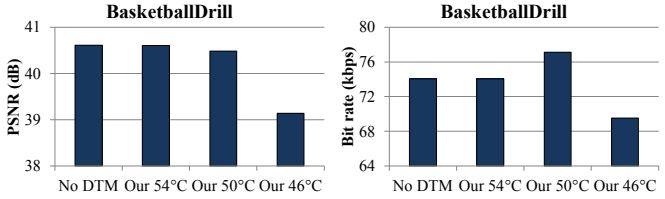
In Fig. 8 (a)-(d), thermal maps of the chip steady temperature when using no DTM algorithm and our DTM with different threshold temperatures (54ºC, 50ºC and 46ºC) for encoding the *RaceHorses* sequence. With these thermal images it is possible to see clearer the impact of using our DTM algorithm in contrast with no DTM in the temperature over the whole chip. Fig. 13 shows the maximum, average and minimum core temperature for the whole *BasketballDrill* sequence encoding considering our DTM with all evaluated thresholds comparing with no DTM. We can see that when using our DTM algorithm the difference between max and min temperature is reduced as the threshold temperature decreases, which also reduces the thermal cycling effects.



(a) no DTM (b) DTM 54 ºC (c) DTM 50 ºC (d) DTM 46 ºC

Fig. 8: Thermal maps of the when encoding the *RaceHorses* sequence.

Fig. 9 shows the impact of our DTM algorithm in terms of PSNR (a) and bit rate (b) in comparison with using no DTM for five video sequences. When using no DTM the encoder chooses the parameters that will provide the best encoding performance, which implies in the final temperature. The degradation of the target attributes increases as the threshold temperature decreases. However, as our algorithm is able to perform appropriate encoder parameter selection, the resulted degradation is low. When the threshold temperature is set to 54ºC, the average PSNR loss is of 0.007 dB while the bit rate slightly increases 0.99% on average for all sequences. When the threshold temperature is set to the lowest value of 46ºC, the degradation is higher no larger than (on average) 1.81 dB for PSNR and 0.84% for the bit rate.



(a) (b)

Fig. 9: PSNR and bit rate results for sequence BasketballDrill.

V. CONCLUSION

For multimedia embedded systems under tight constraints, where temperature, performance, and quality cannot be compromised significantly, we introduced an application-driven dynamic thermal management algorithm for the High Efficiency Video Coding (HEVC) standard. In order to analyze the temperature behavior when encoding different video sequences using HEVC, we performed an extensive thermal analysis using an in-house IR camera based thermal setup. Based on this analysis, we formulate a relationship between temperature and different encoder parameters that was leveraged to develop the proposed DTM algorithm. As results, we illustrate that the dynamic adaptation of encoder parameters is beneficial for managing temperature at the application level while providing negligible video quality loss.

ACKNOWLEDGMENT

This work is supported in parts by the German Research Foundation (DFG) as part of the priority program "Dependable Embedded Systems" (SPP 1500 - spp1500.itec.kit.edu).

REFERENCES

[1] B. Bross, W. J. Han, G. J. Sullivan, J. R. Ohm, T. Wiegand, “High Efficiency Video Coding (HEVC) text specification draft 10”, 2013.

[2] ITU-T, “SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS - Infrastructure of Audiovisual Services - Coding of Moving Video – High Efficiency Video Coding.” Apr. 2013.

[3] D. Marpe, H. Schwarz, S. Bosse, “Video compression using nested quadtree structures, leaf merging, and improved techniques for motion representation and entropy coding”, TCSVT, vol. 20, no. 12, pp. 1676–1687, 2010.

[4] M. T. Pourazad, C. Doutre, M. Azimi, P. Nasiopoulos, “HEVC: The New Gold Standard for Video Compression: How Does HEVC Compare with H.264/AVC?”, IEEE Consumer Electronics Magazine, pp. 36-46, 2012.

[5] M. U. K. Khan, M. Shafique, J. Henkel, “AMBER: Adaptive Energy Management for On-Chip Hybrid Video Memories”, IEEE ICCAD, pp. 405-412, 2013.

[6] M. Shafique, J. Henkel, “Low Power Design of the Next-Generation High Efficiency Video Coding”, ASP-DAC, 2013.

[7] ITRS. http://www.itrs.net, 2013.

[8] J. Henkel, L. Bauer, N. Dutt, P. Gupta, S. Nassif, M. Shafique, M. Tahoori, N. Wehn, “Reliable on-chip systems in the nano-era: Lessons learnt and future trends”, IEEE DAC, 2013.

[9] J. Henkel, T. Ebi, H. Amrouch, H. Khdr, “Thermal management for dependable on-chip systems”, ASP-DAC, pp. 113–118, 2013.

[10] “Failure mechanisms and models for semiconductor devices”, JEDEC publication JEP122C, [Online]. Available: http://www.jedec.org

[11] I. Yeo, C. C. Liu, E. J. Kim, “Predictive dynamic thermal management for multicore systems”, IEEE DAC, pp. 734-739, 2008.

[12] F. Zanini, D.Atienza, G. De Micheli, “A control theory approach for thermal balancing of mpsoc”, ASP-DAC, pp. 37–42, 2009.

[13] T. Ebi, M. Faruque, J. Henkel, “Tape: Thermal-aware agent-based power economy for multi/many-core architectures”, IEEE ICCAD, pp. 302–309, 2009.

[14] D. Brooks, M. Martonosi, “Dynamic thermal management for high-performance microprocessors”, HPCA, pp. 171–182, 2001.

[15] M. D. Powell, M. Gomaa, T. N. Vijaykumar, “Heat-and-run: leveraging smt and cmp to manage power density through the operating system”, ASPLOS, pp. 260–270, 2004.

[16] W. Lee, K. Patel, M. Pedram, “Dynamic thermal management for mpeg-2 decoding”, ISLPED, pp. 316–321, 2006.

[17] W. Lee, K. Patel, M. Pedram, “GOP-Level Dynamic Thermal Management in MPEG-2 Decoding”, IEEE TVLSI, vol. 16, no. 6, pp. 662 – 672, 2008.

[18] A. Mirtar, S. Dey, A. Raghunathan, “Adaptation of video encoding to address dynamic thermal management effects”, IGCC, pp. 1–10, 2012.

[19] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, T. Wedi, “Video coding with H.264/AVC: Tools, Performance, and Complexity”, IEEE Circuits and System Magazine, vol. 4, no. 1, pp. 7-28, 2004.

[20] F.-J. Mesa-Martinez, M. Brown, J. Nayfach-Battilana, J. Renau, “Measuring power and temperature from real processors”, IPDPS, pp. 1–5, 2008.

[21] S. Reda, “Thermal and power characterization of real computing devices”, IEEE JETCAS, vol. 1, no. 2, pp. 76–87, 2011.

[22] C. Lian, M. Knox, K. Sikka, W. Xiaojin, A.J. Weger, “Development of a flexible chip infrared thermal imaging system for product qualification”, SEMI-THERM, pp. 337–343, 2012.

[23] “DIAS Infrared Camera”, http://www.dias-infrared.com/pdf/pyroview380lcompact eng.pdf.

[24] Joint Collaborative Team on Video Coding (JCT-VC), “HM 11.0 Reference Software” [Online]. Available: http://hevc.hhi.fraunhofer.de/

[25] M. Shafique, B. Zatt, J. Henkel, “A complexity reduction scheme with adaptive search direction and mode elimination for multiview video coding”, PCS, pp. 105–108, 2012.

[26] Chair for Embedded Systems, KIT, Germany, http://ces.itec.kit.edu.