



Delft University of Technology

ALMARVI System Solution for Image and Video Processing in Healthcare, Surveillance and Mobile Applications

Al-Ars, Zaid; van der Vlugt, Steven; Jääskeläinen, Pekka; van der Linden, Frank

DOI

[10.1007/s11265-018-1423-2](https://doi.org/10.1007/s11265-018-1423-2)

Publication date

2019

Document Version

Final published version

Published in

Journal of Signal Processing Systems

Citation (APA)

Al-Ars, Z., van der Vlugt, S., Jääskeläinen, P., & van der Linden, F. (2019). ALMARVI System Solution for Image and Video Processing in Healthcare, Surveillance and Mobile Applications. *Journal of Signal Processing Systems*, 91(1), 1-7. <https://doi.org/10.1007/s11265-018-1423-2>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.



ALMARVI System Solution for Image and Video Processing in Healthcare, Surveillance and Mobile Applications

Zaid Al-Ars¹ · Steven van der Vlugt² · Pekka Jääskeläinen³ · Frank van der Linden²

Published online: 20 November 2018

© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

ALMARVI is a collaborative European research project funded by Artemis involving 16 industrial as well as academic partners across 4 countries, working together to address various computational challenges in image and video processing in 3 application domains: healthcare, surveillance and mobile. This paper is an editorial for a special issue discussing the integrated system created by the partners to serve as a cross-domain solution for the project. The paper also introduces the partner articles published in this special issue to discuss the various technological developments achieved within ALMARVI spanning all system layers, from hardware to applications. We illustrate the challenges faced within the project based on use cases from the three targeted application domains, and how these can address the 4 main project objectives addressing 4 challenges faced by high performance image and video processing systems: massive data rate, low power consumption, composability and robustness. We present a system stack composed of algorithms, design frameworks and platforms as a solution to these challenges. Finally, the use cases from the three different application domains are mapped on the system stack solution and are evaluated based on their performance for each of the 4 ALMARVI objectives.

1 Introduction

ALMARVI is a European project (funded by Artemis with grant no. 621439) that aims at providing cross-domain many-core platform solution, system software stack, tool chain, and adaptive algorithms that enable massive data-rate image and video processing with high energy efficiency. ALMARVI provides mechanisms and support for high degree of adaptability at various system layers that abstracts the variations in the underlying platforms (e.g., to enable portability across of-the-shelf components), communication channels (e.g., available bandwidth), application behaviour

(e.g., dynamic workloads, changing requirements) from the application developer. This is crucial for providing consistent performance efficiency in an interoperable manner when considering heterogeneous platform options and dynamic operating conditions. The key is to leverage image and video content-specific properties, application-specific features, and inherent resilience properties of image and video processing applications.

This paper serves as an editorial for the ALMARVI special issue discussing the various technological developments achieved by project partners at different system layers. The special issue is a collection of articles contributions provided as a collaborative effort by multiple project partners regarding their specific technology.

In order to identify the relationships between the article contributions and how they relate to the ALMARVI project, this paper provides an overview of the project and its system layers. In addition, we show how the technological developments together can fulfill the objectives of the project in the 3 project application domains: healthcare, surveillance and mobile.

In each application domain, a number of ALMARVI demonstrators are considered. To demonstrate and validate project developments and results, ALMARVI technologies will be evaluated using the following demonstrators: 1) Healthcare demonstrators: medical imaging assisted diagnosis with a mixture of real-time and non real-time

✉ Zaid Al-Ars
z.al-ars@tudelft.nl

Steven van der Vlugt
steven.van.der.vlugt@philips.com

Pekka Jääskeläinen
pekka.jaaskelainen@tut.fi

Frank van der Linden
frank.van.der.linden@philips.com

¹ Delft University of Technology, Delft, The Netherlands

² Philips Healthcare, Best, The Netherlands

³ Tampere University of Technology, Tampere, Finland

image processing tasks for different healthcare applications like minimal invasive treatment for cardiovascular diseases. 2) Surveillance demonstrators: distributed monitoring using mobile and fixed video sensor nodes as well as continuous monitoring of industrial processes. 3) Mobile demonstrators: novel ultra energy-efficient high performance heterogeneous multicore platform for mobile applications and nomadic embedded devices of the future.

For each domain, the common researched technologies for algorithms, design tools, system software stack, and many-core execution platforms are described. We also show how the various demonstrators achieve their target requirements. The investigations of the various partners in the ALMARVI project target the following specific objectives of the ALMARVI project: 1. massive data rate, 2. low power, 3. composability, 4. robustness.

This paper is organized as follows. Section 2 describes the various abstraction layers of the ALMARVI system stack. We also present the different articles published in this special issue and how they fit in these abstraction layers of the system stack. Section 3 lists the articles included in this special issue and how they relate to the ALMARVI system stack. Section 4 explains the mapping of the demonstrators of each ALMARVI partner on the ALMARVI system stack and discusses the details of the various demonstrators and their achieved results.

The conclusions are presented in Section 5.

2 System Stack

The ALMARVI project has a diverse range of applications spanning multiple application domains in healthcare, surveillance and mobile. In order to reduce the effort of solving the image and video processing challenges in these domains, the partners worked on an integrated ALMARVI system stack that helps them share solutions across the three application domains. This section discusses the various layers of the stack and the technologies used to create it.

Figure 1 shows the ALMARVI system stack solution, identifying three different layers with their associated technologies. These different layers help in abstracting away

the complexities of the technologies at each layer, and allow for reusing these technologies transparently across the 3 application domains on top of the stack. These three layers are as follows.

The software layer is the top layer, where two types of algorithms are used in the project. The first type consists of existing algorithms already known to solve a specific image and video processing problem that have been used off-the-shelf. The second type consists of algorithms that some partners developed within the project to satisfy the needs of their ALMARVI use case. In addition to the actual algorithms, this layer is concerned with the tools and techniques developed and applied to tune the algorithms to specific platform architectures.

The middleware layer is the system abstraction layer, where partners experimented with abstraction methods to allow for making the solutions portable and reusable. For ALMARVI, OpenCL has been used as a common abstraction layer due to its wide support from the industry, its vendor-independent character, and due to the availability of well-developed OpenCL interfaces for various hardware platforms. On the other hand, some ALMARVI partners also used platform-specific abstractions developed for the specific partner solution. Some partners implemented both OpenCL as well as platform-specific abstractions and compared to each other.

The hardware layer is the execution platform layer, for which partners used two types of hardware processors. The first type is represented by off-the-shelf platforms, which is based on commercially available processors from independent vendors such as ARM, Intel, NVidia, etc. The second type consists of dedicated ALMARVI platforms created by the various partners of the project. These platforms are: 1. the rVEX dynamically adaptive core created by TUDelft (Delft University of Technology) [2, 3], 2. the TCE customizable processor tool chain created by TUT (Tampere University of Technology) [4], 3. the edkDSP accelerated hardware processors created by UTIA (Institute of Information Theory and Automation) [5].

Figure 1 The different abstraction layers of the ALMARVI system stack solution.

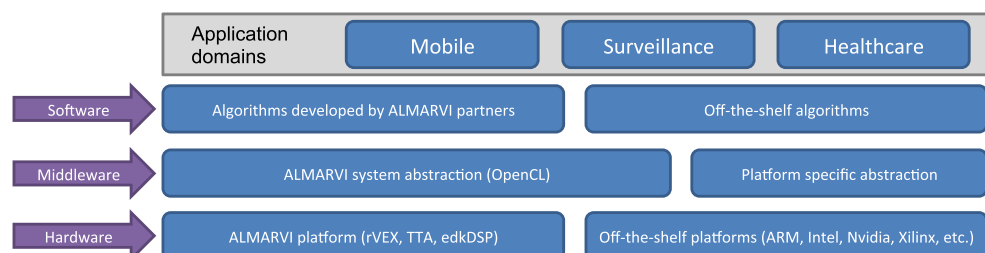
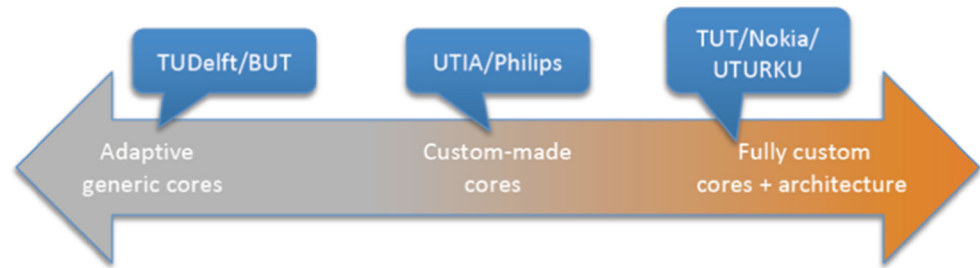


Figure 2 Spectrum of design space alternatives for the ALMARVI processing platforms used by project partners.



The hardware processing platforms used by the ALMARVI partners span a wide spectrum of design space alternatives as shown in Figure 2. At the very left, adaptive generic rVEX softcores are used by the TUDelft and BUT (Brno University of Technology). At the very right, customized TCE cores and architectures targeting ASIC implementation are used by TUT, Nokia and UTURKU (University of Turku). As a trade off between these two, more customizable cores for the specific applications are used by UTIA (edkDSP) and Philips Healthcare (FPGA accelerators).

3 Special Issue Articles

This special issue includes a number of articles that describe the various technological developments realized within the ALMARVI project. The articles are listed below, and are classified depending on the system stack layer they concern (see Figure 1).

The software layer —Articles about algorithm design and implementation

1. “*Image Restoration in Portable Devices: Algorithms and Optimization*”, by Jan Kamenicky, Filip Sroubek, Barbara Zitova, Jari Hannuksela and Markus Turtinen. This article discusses new innovative image and video processing algorithms focusing on denoising. It proposes several parallel implementations and simplifications of these algorithms to increase their performance and reduce their power consumption.
2. “*Monotonic Optimization of Dataflow Buffer Sizes*”, by Martijn Hendriks, Hadi Alizadeh Ara, Marc Geilen, Twan Basten, Ruben Guerra Marin, Rob de Jong, and Steven van der Vlucht. This article discusses design optimization methods carried out on video-processing algorithms to allow for an optimal tradeoff between system throughput and the sizes of buffers in the design.

The middleware layer —Articles about system abstraction and algorithm portability

3. “*Exploiting Task Parallelism with OpenCL: A Case Study*”, by Pekka Jääskeläinen, Ville Korhonen, Matias Koskela, Jarmo Takala, Karen Egiazarian, Aram Danielyan, Cristóvão Cruz, James Price and Simon McIntosh-Smith. This article discusses using OpenCL to describe task-level parallelism in algorithms, and how such a description can be used to enable higher performance of applications.
4. “*Frame-based Programming, Stream-based Processing for Medical Image Processing Applications*”, by Joost Hoozemans, Rob de Jong, Steven van der Vlucht, Jeroen Van Straten, Uttam Kumar Elango and Zaid Al-Ars. This article discusses methods to deploy image and video processing pipelines that are developed frame-oriented on a hardware platform that is stream-oriented, such as an FPGA. Such methods simplify the effort of programming specialized hardware platforms.

The hardware layer —Articles about processing hardware platform and core design

5. “*ALMARVI Execution Platform: Heterogeneous Video Processing SoC Platform on FPGA*”, by Joost Hoozemans, Jeroen van Straten, Timo Viitanen, Alekski Tervo, Jiri Kadlec and Zaid Al-Ars. This article discusses the ALMARVI integrated video processing hardware platform used by the various project partners to run their respective algorithms.
6. “*Modeling and Analysis of FPGA Accelerators for Real-time Streaming Video Processing in the Healthcare Domain*”, by Steven van der Vlucht, Hadi Alizadeh Ara, Rob de Jong, Martijn Hendriks, Ruben Guerra Marin, Marc Geilen and Dip Goswami. This article uses a model-driven analysis and detailed hardware level simulation to optimize the FPGA implementation of streaming video processing algorithms.
7. “*Tools and Techniques for Implementation of Real-time Video Processing Algorithms*”, by Vecdi Emre Levent, Aydin E. Guzel, Mustafa Tosun, Mert Buyukmichi, Furkan Aydin, Sezer Goren, Cengiz Erbas, Toygar Akgun and H. Fatih Ugurdag. This article discusses tools and techniques to enable rapid FPGA-based hardware design that allows fast design space exploration.

4 Demonstrators

In this section, we discuss how the different ALMARVI demonstrators are mapped on to the ALMARVI system stack [6]. We have experimented with both off-the-shelf solutions as well as the solutions developed within ALMARVI. For off-the-shelf solutions we evaluate the current state of the art tools and techniques. For the ALMARVI solutions, we evaluate the results compared to, if available, the current state of the art tools and techniques, or evaluate how well the developed solution satisfies project objectives.

Figure 3 shows the mapping of the different demonstrators to the ALMARVI system stack listed per partner. For each demonstrator, the left column (light gray) indicates the ALMARVI solution, the right column (dark gray) indicates the off-the-shelf software (i.e., algorithms), middleware (i.e., abstraction) and hardware (i.e., platform). For example, the first demonstrator to the left of the figure is the interventional x-ray system developed by Philips, which uses off-the-shelf software/algorithms, and combines this with both off-the-shelf as well as ALMARVI-specific middleware abstractions and hardware platforms. VTT and VISIDON each worked on the optimization of ALMARVI-specific algorithms (power/quality trade off) targeting off-the-shelf hardware (CPU/GPU) through off-the-shelf and ALMARVI middleware. UEF (University of Eastern Finland) has worked both with existing algorithms as with ALMARVI optimized algorithms through OpenCL and off-the-shelf abstraction to ALMARVI (rVex, TTA) and off-the-shelf platforms. Aselsan and Nokia have demonstrated the portability of already existing algorithms as well as ALMARVI optimized algorithms through off-the-shelf abstraction to both off-the-shelf and ALMARVI platforms. CAMEA (together with BUT) created ALMARVI-optimized algorithms for road traffic surveillance and used vendor specific tools to come to deploy on an off-the-shelf platform. Hurja (the smallest partner in the consortium) has implemented a fully off-the-shelf solution, consisting of existing algorithms with platform specific abstraction on an off-the-shelf platform. These examples of various system stack mappings clearly show the differences in the needs of each partner depending of the requirements of their respective applications.

4.1 Healthcare Demonstrators

The healthcare demonstrators use acceleration fabrics and exploitation of parallelism to achieve image processing at the required performance level. They demonstrate software portability to various platforms through the use of high level design tools.

The Philips demonstrators are related to a real-time video processing pipeline of an interventional x-ray system. The research questions, objectives, baseline and challenges were addressed with three different demonstrators each addressing several of the challenges. In order to investigate alternative solutions that satisfy system requirements, a complex C++ video processing algorithm was implemented both on an FPGA as well as on GPU using the OpenCL language. The demonstrator outcomes related to FPGA tooling are currently being used by Philips for product development. Follow-up studies were started to further address the gap between OpenCL and FPGA development in order to take benefit of the abstraction offered by higher level languages.

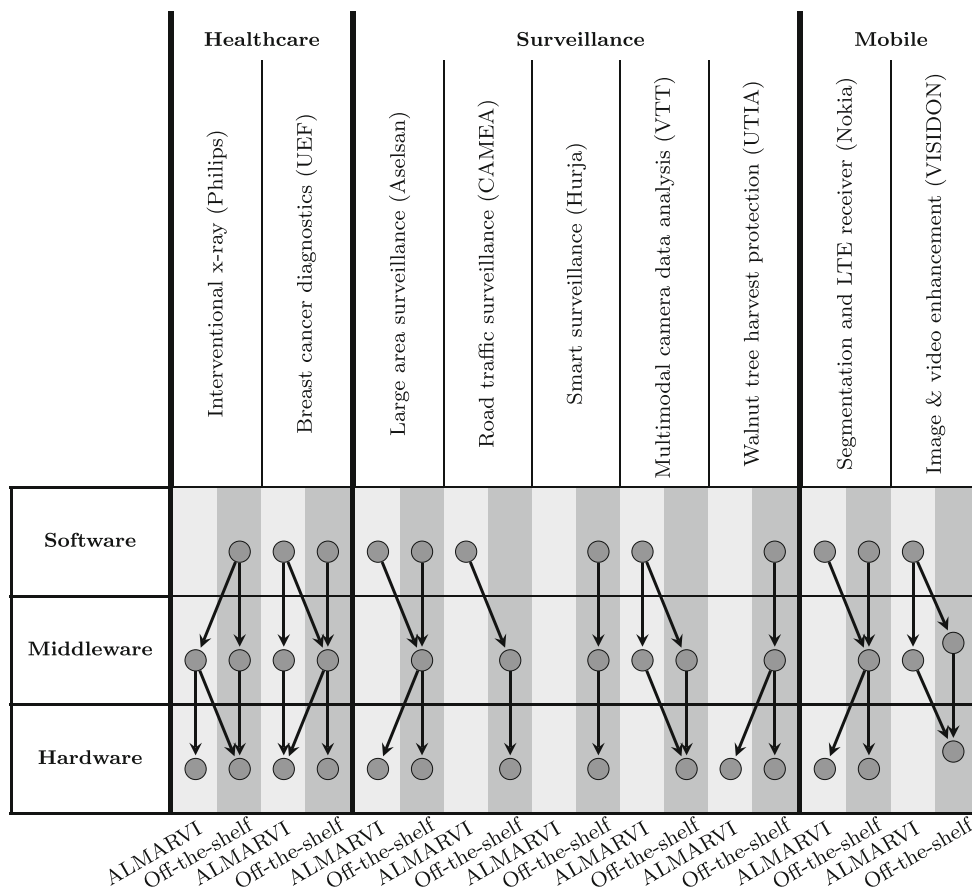
The breast cancer diagnostics demonstrator from UEF is a demo system for conducting image analysis experiments on histopathological breast cancer images. The system is used to experiment with feature extraction, segmentation, and classification methods for medical imagery. Using off-the-shelf hardware three versions of the demo system were implemented and compared: a Matlab-based application; a stand-alone C-based application targeting CPUs and the rVEX ALMARVI platform; and an efficient OpenCL-based application targeting GPUs. The underlying challenges were to create an efficient parallel implementation of the Adaptive Bottom-Hat Filtering algorithm.

4.2 Security Demonstrators

In the security domain, ALMARVI has developed five main demonstrators: large area video surveillance, road traffic surveillance, smart surveillance, analysis of multimodal camera data and the protection of walnut tree harvest against birds.

The large area video surveillance demonstrator from Aselsan is based on a prototype security setup that consists of two outdoor daylight IP cameras with processing boards and a desktop computer acting as command centre. End node processing (very close to the IP cameras) consists of optical flow and segmentation. The algorithms executed on the command centre (desktop computer) are *optical flow*, segmentation and *image fusion*. As a result, three different sub-demonstrators were built to address these three different algorithms. The optical flow demonstrator was successfully implemented on GPU and FPGA, the latter was developed with the MAFURES tool [1]. The optical flow demo showed that with FPGA we can achieve lower power consumption and smaller form factor for a wide range of resolutions and throughput values (up to 75 GFLOPS). The segmentation demo was built on an embedded platform with a built-in GPU. This demo is to be used for detailed image analysis run on server farms once the optical flow demo dispatches each video frame for further processing. For segmentation,

Figure 3 Demonstrator mapping on the ALMARVI system stack. For each demonstrator, the left column (light gray) indicates the ALMARVI solutions, the right column (dark gray) indicates the off-the-shelf solutions.



Aselsan showed that an embedded platform with built-in GPU is able to meet the performance per Watt per dollar requirements. For example, for segmentation, 5 fps at 640×480 pix/f (1.5 Mpix/s) was achieved on a COM Express embedded computer (VXG101 of Connect Tech). For the image fusion demo, the same has been demonstrated with performances up to 172 Mpix/s.

The road traffic surveillance system from CAMEA has to operate at locations where no external power supply is available. Therefore, we need to ensure low power consumption of system (keeping the real-time processing requirements) to conserve battery power. Suitable low power hardware solution is Xilinx Zynq combining an ARM processor with programmable logic that allows design of low power accelerators. This demonstrator includes real-time object detection in HD video using a low cost and low power compact system. Such embedded system will be used for road traffic surveillance applications such as counting vehicles, license plate detection (for further recognition) and many others. The core of the system (the object detector) is based on the Waldboost algorithm and is trained for car license plates based on trained classifiers, which was done together with BUT. The object detection is fully hardware accelerated in FPGA and is performed by the IP core of

Waldboost based detector (developed by BUT). Results of the detection are processed by ARM cores of Xilinx Zynq which are also setting up and driving the CMOS sensor (Python 2k CMOS from On Semiconductor). The camera is also equipped with an H.264 hardware encoder from Fujitsu. Resultant camera output enhanced by detection results is compressed and streamed in MPEG-TS over a wi-fi and Ethernet network and can be displayed by most media players with H.264 decoding support.

The smart surveillance demonstrator from Hurja uses a physical prototype module which has two built in cameras, removable storage and a Raspberry Pi2. The cameras face away from each other to capture video from both sides of the module. The application is to monitor people entering and leaving a designated area. We managed to build a pedestrian detection and counting system with off-the-shelf components. The application running on the Raspberry Pi consists of an OpenCV based algorithm that identifies areas of interest (i.e. large moving objects) in the images. We came to the conclusion that by using a pre-trained classifier running on an external server to combine all the detection results, it is possible to count multiple unique pedestrians across multiple camera systems by running the same basic detection algorithm.

The multi-camera object tracking demonstrator from VTT contributes to enabling massive data rate processing by utilizing a heterogeneous and scalable execution platform in the surveillance context, and supporting an application specific architecture for video processing. The video processing algorithms also support parallelization. The demonstrator focuses on multi-camera object recognition and tracking in cameras and between cameras to get 3D context awareness of the person's motions. The demonstrator utilizes many camera nodes that are connected to a central processing unit, and therefore presents distributed and parallel processing. The demonstrator presents algorithms such as automatic calibration to get sense of 3D space, and person recognition and tracking. The challenge is to get the 3D sense and object recognition tracking accurate enough and tackling performance challenge of integration of many video streams. The demonstrator utilizes many camera nodes that are connected to a central processing unit, and presents algorithms such as automatic calibration to get sense of 3D space, and person recognition and tracking. The demonstrator is up and running, a 3D sense is created, and persons are identified and tracked from one camera to another.

The walnut harvest protection demonstrator from UTIA is focused on demonstration of the hardware capabilities of the platform developed by UTIA. The steps implemented include migration of the execution platform to the Zynq7030 FPGA device which contains more programmable logic resources. Building block algorithms for the demo were implemented and demonstrated on new hardware with the SDSoC tooling and OpenCV libraries. The demo includes motion detection, background subtraction, and object detection.

4.3 Mobile Demonstrators

Nokia has demonstrated the benefits and capabilities of targeting many heterogeneous acceleration fabrics: CPUs, DSPs, GPUs and FPGAs from a single source code. During the project, we proposed and implemented a baseline of using different computation frameworks such as TTA and rVex under a common acceleration API through pocl. We have successfully demonstrated the application of the ALMARVI toolset to implement application specific cores to different ends of the computation complexity spectrum. From one end, we target low latency but extreme throughput image and radio communications, while from the other we target ultra-low power budget audio computations. The toolsets used in this demonstrator can be applied across domains. Two different demonstrators were further developed: MIMO detection for high throughput and ultra-low power audio signal processing.

The image and video enhancement demonstrator from VISIDON focuses on implementation of image enhancement algorithms for a standard off-the-shelf mobile platform. The main purpose of the demonstrator is to find performance and power consumption differences between various heterogeneous processing units available on the platform, namely ARM CPU, ARM NEON, and GPU. Image de-noising software implemented and optimized for different processing units is used for experimenting with processing speed and power. In addition, design and software portability related to ALMARVI is considered. These requirements are addressed in one demonstrator. This demonstrator considered a mobile image enhancement use case, where we studied how to efficiently use heterogeneous mobile platforms for image de-noising. The demonstrator was implemented on a Qualcomm Snapdragon platform running Android OS. The performance of the demonstrator was analyzed based on the objectives set in the early phase of the project.

5 Conclusion

We discussed how the different partners of the ALMARVI project worked to fulfill the overall objective of the project, which is to provide solutions to enable massive data rate image/video processing at low power budgets under variability conditions. The partners achieved this with the integrated and rigorous evaluation of all their use cases for performance and power consumption. In addition, an analysis and evaluation is given on the ALMARVI demonstrators in the different application domains against the overall project objectives. The various ALMARVI use cases from the three application domains (healthcare, surveillance and mobile) are considered. To demonstrate and validate the project developments and results, the ALMARVI results were evaluated using demonstrators from the three different application domains. For each domain, the common researched concepts for algorithms, design tools, system software stack, and many-core execution platforms are described. We discussed that the various demonstrators collectively used all of the system stack components, in each of the application domains. At the hardware layer, partners compared off-the-shelf hardware solutions to ALMARVI-specific processors, such as the rVEX, TTA or edkDSP. At the middleware layer, partners experimented with portable solutions, such as OpenCL, and compared it to their solution-specific abstraction. At the software layer, several partners developed their own algorithms and optimization techniques to satisfy the needs of their ALMARVI use case. We also showed how the various demonstrators achieve their target requirements and collectively cover the objectives of the project.

Acknowledgements This work has been supported by the ALMARVI European Artemis project no. 621439.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Guzel, A.E., Levent, V.E., Tosun, M., Özkan, M.A., Akgun, T., Büyükkaydin, D., Erbas, C., Ugurdag, H.F. (2016). Using high-level synthesis for rapid design of video processing pipes. In *2016 IEEE East-West design test symposium (EWDTS)* (pp. 1–4). <https://doi.org/10.1109/EWDTS.2016.7807644>.
2. Hoozemans, J., Wong, S., Al-Ars, Z. (2015). Using vliw softcore processors for image processing applications. In *Proc. International conference on embedded computer systems: architectures, modeling and simulation, Samos, Greece*.
3. Hoozemans, J., Heij, R., van Straten, J., Al-Ars, Z. (2017). *VLIW-Based FPGA computation fabric with streaming memory hierarchy for medical imaging applications*, (pp. 36–43). Cham: Springer International Publishing.
4. Jääskeläinen, P., Viitanen, T., Takala, J., Berg, H. (2017). *HW/SW Co-design Toolset for Customization of Exposed Datapath Processors*, (pp. 147–164). Berlin: Springer International Publishing.
5. Kadlec, J. (2015). Video chain demonstrator on Xilinx Kintex7 FPGA with EdkDSP floating point accelerators. In *2015 International conference on embedded computer systems: architectures, modeling, and simulation (SAMOS), IEEE* (pp. 310–314).
6. van der Vlugt, S., Haataja, K., Toivanen, P., Braithwaite, B., Akgün, T., Uğurdağ, H.F., Levent, V.E., MaršCsík, L., Väänänen, A., Härkönen, V., Keränen, J., Pohl, Z., Berg, H., Turtinen, M., Al-Ars, Z. (2017). Evaluation of the almarvi demonstrators. Online, http://almarvi.eu/assets/d5.7-evaluation-of-the-almarvi-demonstrators_v1.0.pdf.