Design and management of image processing pipelines within CPS: acquired experience towards the end of the FitOptiVis ECSEL Project

Carlo Sau[1], Claudia Rinaldi[2], Luigi Pomante[2], Francesca Palumbo[3], Giacomo Valente[2], Tiziana Fanni[3], Marcos Martinez[4], Frank van der Linden[5], Twan Basten[6], Marc Geilen[6], Geran Peeren[5], Jiří Kadlec[7], Pekka Jääskeläinen[8], Lubomír Bulej[9], Francisco Barranco[10], Jukka Saarinen[11], Tero Säntti[12], Maria Katiuscia Zedda[13], Victor Sanchez[6], Shayan Tabatabaei Nikkhah[6], Dip Goswami[6], Guillermo Amat[14], Lukáš Maršík[15], Mark van Helvoort[5], Luis Medina[16], Zaid Al-Ars[17], Ad de Beer[5]

[1]Università degli Studi di Cagliari, [2]Università degli Studi dell'Aquila, [3]Università degli Studi di Sassari, [4]Thales Alenia Space España, [5]Philips, [6]Eindhoven University of Technology, [7]Institute of Information Theory and Automation, [8]Tampere University, [9]Charles University, [10]Universidad de Granada, [11]Nokia, [12]University of Turku, [13]Abinsula, [14]InstitutoTecnólogico de Informática, [15]Camea, [16]Seven Solutions, [17]Delft University of Technology

***Abstract*— Cyber-Physical Systems (CPSs) are dynamic and reactive systems interacting with processes, environment and, sometimes, humans. They are often distributed with sensors and actuators, characterized for being smart, adaptive, predictive and react in real-time. Indeed, image- and video-processing pipelines are a prime source for environmental information for systems allowing them to take better decisions according to what they see. Therefore, in FitOptiVis, we are developing novel methods and tools to integrate complex image- and video-processing pipelines. FitOptiVis aims to deliver a reference architecture for describing and optimizing quality and resource management for imaging and video pipelines in CPSs both at design- and run-time. The architecture is concretized in low-power, high-performance, smart components, and in methods and tools for combined design-time and run-time multi-objective optimization and adaptation within system and environment constraints.**

Keywords*— *Image processing, video processing, distributed system, heterogeneous system, multi-objective optimization, cyber-physical systems

I. INTRODUCTION

Smart systems integration is one of the essential capabilities required to improve the competitiveness of European industry in the ECSEL application domains. This is especially relevant for Cyber-Physical Systems (CPSs), i.e., autonomous distributed integrations of electronic systems and software (SW), tightly coupled to and interacting with physical systems and their environment. The Electronic Components and Systems for European Leadership (ECSEL) project FitOptiVis (*From the cloud to the edge - smart IntegraTion and OPtimisation Technologies for highly efficient Image- and Video-processing Systems*) [1]~~[2]~~ deals with novel design and run-time approaches for imaging and video pipelines in CPSs. Images play a central role in human perception and in the understanding of our environment. Accordingly, CPSs need visual context and awareness to take correct decisions and to perform appropriate actions. Fig. 1 shows a generic distributed image- and video-processing pipeline for CPSs. Advanced image- and video-processing is computing intensive, whereas for adequate behavior results need to be available with low latencies and high throughput. Devices often need to operate with low energy and limited heat dissipation, while optimization for other qualities may be also important. FitOptiVis started on June 1st, 2018 and its duration, originally 3 years, has been extended in November 2020, due to COVID-19 reasons which affected integration activities, for 6 additional months. In the following, we highlight project objectives, explain the selected approach, describe target use cases (UCs), and present the main results obtained after more than two and a half years of activities. Please note that this is an extension of the work presented in [2], where an overview of the project and its main progress after the first two years have been shown. With respect to such a previous work, in this case, besides reporting six more months of experience, we provided a deeper dive into the ten project use cases, which are the driver of technology development. In particular, we here highlight the use case state of the art, faced challenges, and new solutions enabled by FitOptiVis technologies.

II. FitOptiVis Context

CPSs are systems that are in feedback with their environment, possibly with humans in the loop. They are often distributed, involve sensors and actuators. In terms of properties, they are often required to be smart, adaptive, and predictive, and real-time reactive [3].

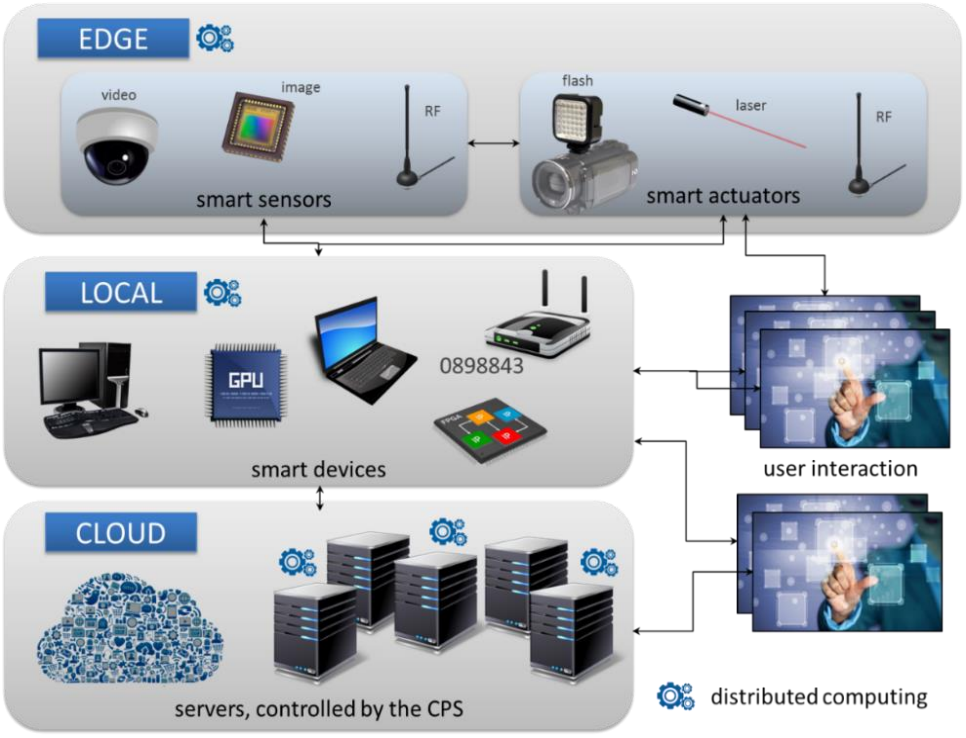

Fig. 1. Generic configuration of imaging and video pipelines in CPSs.

Image- and video-processing pipelines are a prime source for environmental information improving the possibilities of active, relevant feedback. They often need to satisfy stringent non-functional constraints (e.g., performance, energy), so advanced image- and video-applications become very complex. The main objective of FitOptiVis is to develop an approach for smart integration of image- and video-processing pipelines for distributed CPSs applying a combined design- and run-time multi-objective optimization of quality and resource usage under system and environment constraints. This is supported by a reference architecture, low-power high-performance smart devices, and proper design-time and run-time methodologies and tools. From the implementation point of view, distributed pipelines (see Fig. 1) consist of a heterogeneous configuration of legacy devices, state of the art multi-vendor devices and components, and newly developed application-specific ones. Smart systems integration for image- and video-applications must be built evolutionarily upon earlier developments, and the CPSs must be able to cope with individual components, hardware (HW) and SW, upgrades during their lifetime.

The FitOptiVis objective is being reached by pushing the state of the art in development and run-time support of distributed image- and video-pipelines, targeting primarily real-time performance and energy usage. FitOptiVis is working on providing a reference architecture that enables the integration of state of the art technology and new developments in the mentioned domain, supporting composability built on suitable abstractions of components, embedded sensing, actuation, and processing devices. The reference architecture supports design portability, online multi-objective quality and resource optimization, and run-time adaptation, guaranteeing system constraints and requirements through platform virtualization. A cloud in the FitOptiVis context is a set of connected servers under the control of the CPS. Non-functional aspects other than performance and energy, such as reliability and security, are considered to meet UC-specific objectives but are not an explicit target of research and development.

Design- and run-time models provide a suitable set of component abstractions for performance and energy related to the distributed system configurations, and for the use of processing, communication, and storage resources. A Domain Specific

Language (DSL), the Quality and Resource Management Language (QRML) [4][5], is proposed with well-defined mathematical semantics [6]. QRML allows to specify the modular structure of a CPS with the essential qualities of interest (performance, energy, visual quality, and so on) and the required resource budgets (processing, memory, bandwidth) of video and image tasks, devices, and components. Specific design-time methods and tools address performance and energy optimization as well as seamless, compositional integration of the image- and video-pipelines. The reference architecture provides templates for a flexible virtual platform built on the component abstractions. The image- and video-pipelines can be designed targeting virtual platforms in various configurations corresponding to different points in the multi-objective performance, energy usage, and resource cost space. These virtual platform configurations can be mapped at run-time onto physical resources, depending on their availability and on the needs of other applications. Energy-efficient, high-performance, smart devices and components are developed to support and demonstrate the reference architecture. The developed tools ensure effective resource-usage predictions and simulations for design-space exploration at design-time, and multi-objective optimization during run-time.

FitOptiVis-compliant systems integrate ultra-low power and high-performance devices and components into the realization of the reference architecture. It exploits the advantages of distributed resources, where off-the-shelf processing elements are supported by efficient companion computing elements close to the sensors or actuators (*edge computing*) through, for example, configurable HW accelerators. By applying novel ultra-low power technologies, and a complex, multi-source functionality, which typically needs to run on multiple heterogeneous components, FitOptiVis-compliant systems can meet CPSs needs and to provide the desired properties. They are able (i) to support the adaptation of the complete image or video pipeline, (ii) to ensure dynamic behaviors, and (iii) to guarantee optimal results in terms of power and performance, even in the case the execution-condition changes due to internal or external triggers. Finally, FitOptiVis is leading to shorter development times and improved products with richer functionalities.

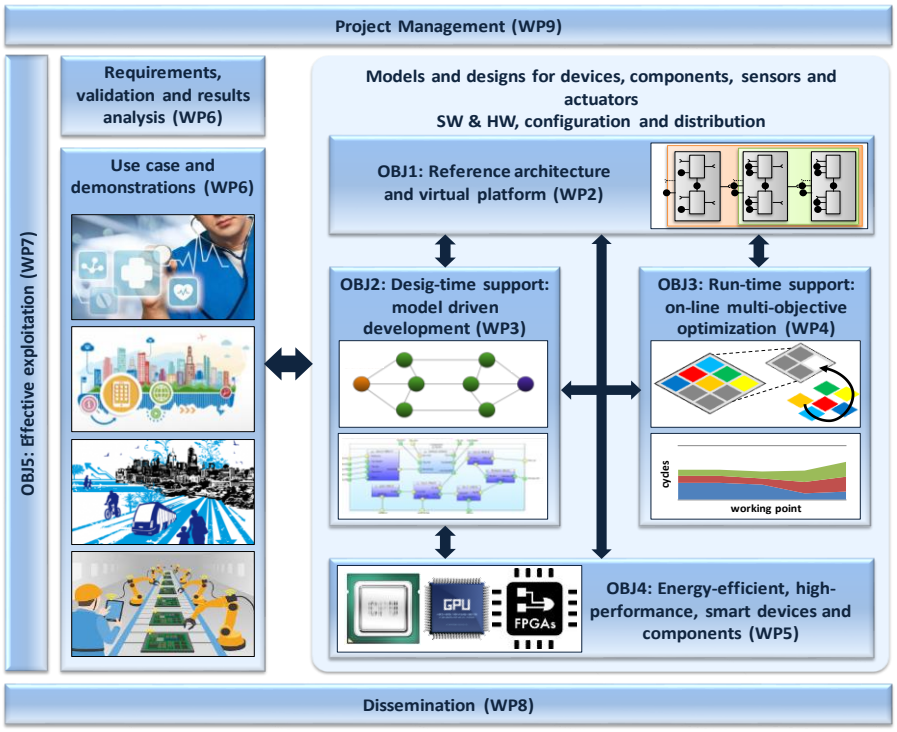

Fig. 2. Organizational approach, dependencies, and operational objectives (OBJ1-5).

III. FitOptiVis Concept, Approach, and Operational Objectives

FitOptiVis developments are driven by industrial UCs that serve as the basis for requirements, demonstration, and validation. Models and abstractions play a crucial role for integrated reference architecture, design methods and run-time operation of CPS image- and video-pipelines. At the moment, all UCs are using QRML [6] to describe the resource options and constraints. Devices and components expose their functional and performance set points, with the corresponding resource requirements,

through minimal interfaces. Model-driven design methods aim to design set points that trade-off functional and performance capabilities against resource usage, for maximal flexibility [7].

Resource virtualization and predictable, composable reconfiguration enable modular, scalable run-time multi-objective adaptation and optimization ensuring quality and real-time performance, even across configurations. Online monitoring techniques [8] enable the evolution of set points and their resource requirements in changing contexts, both operational environments and use patterns. The common reference architecture captures the essential aspects of the envisioned approach in appropriate component abstractions, virtualization techniques and quality- and resource-management protocols. The approach is grounded in the development of smart, high-performance, energy-efficient devices and components to validate concepts and exploit results of FitOptiVis. In summary, the operational objectives of FitOptiVis with the corresponding target can be summarized as:

- *OBJ1: Reference architecture and virtual platform* – The aim is to find template solutions for: component abstractions (covering video and imaging tasks and heterogeneous processing, storage and network devices and components); virtualization supporting scalability, portability, and composability principles; quality and resource management (support for run-time decision making, adaptation, (re-) distribution and upgrades).
- *OBJ2: Design-time support: model driven development* – It is intended to apply performance and energy models, as well as image- and video-processing models; to perform model combinations, multi-objective optimization for performance and energy, co-simulation, design-space exploration, HW/SW co-design targeting Field Programmable Gate Arrays (FPGAs) and custom accelerators; to integrate methods for heterogeneous devices; to perform evolutionary development.
- *OBJ3*: *Run-time support: online multi-objective optimization* – It is about real-time multi-objective combinatorial optimization, data and process distribution, run-time adaptation using virtualization, run-time quality- and resource-management, energy driven adaptations, workload (re-)distribution, and support for run-time upgrades.
- *OBJ4: Energy-efficient, high-performance, smart devices and components* – The focus is on the development of sensors, actuators, communication and processing components, programmable and non-programmable accelerators, ultra-low-power devices.
- *OBJ5: Effective impact* – The goal is to ensure exploitation of the project results in distributed image- and video-processing CPSs and their development.

Fig. 2 summarizes the operational objectives and shows how these are embedded in the organizational approach of FitOptiVis. On the one side, applications, composable entities providing a functionality to a user, are abstracted. On the other side execution platforms, again intended as composable entities but with the capability of running an application, is abstracted or rather virtualized. Abstract applications and virtual execution platforms are part of the reference architecture: applications require budgets to be run, and platforms provide budgets to run applications. Moreover, platforms take parameters and have qualities, such as energy consumption or processing accuracy, which can change according to the specific component and to the adopted set point. Application-to-platform mapping is then performed: besides matching budget demand-provision, if set points are available, e.g., providing different trade-offs in terms of resources and/or qualities, they have to be efficiently managed according to the specific context requirements. Fig. 3 sketches the described technical approach.

The FitOptiVis operational objectives (OBJ1-5) are motivated by industrial UCs that share demanding image- and video-processing, while differing in other aspects like available processing power, energy budgets, safety requirements, configuration distribution, configuration dynamics, etc. The UCs dictate the requirements for integral development approach and smart integration of image- and video-processing pipelines for CPSs. UCs also form the basis for demonstrators. The demonstrators serve for validation and analysis of the project results, and for their exploitation. The UCs are detailed in Section V. Due to the industry-driven approach, technical specifications of the FitOptiVis platform, components and tools come from real needs. Both cross-domain and domain-specific requirements are considered to cover all the necessary design set points.

UCs and demonstrators have been chosen to have diversity in image- and video-processing tasks. The different application domains are expected to benefit from the common reference architecture, design methods and run-time approach, and, generally, from re-usable and easily customizable devices and components. Generic and unified platform and design methods enable the quick creation of image- and video-processing infrastructure in different domains, and simplified connections among systems across domains. This will become increasingly important in our rapidly evolving society where systems are without exception

connected on the *Internet of Things*. At the end of FitOptiVis we intend to deliver generic results applicable in many different domains, even outside the scope of the project itself, and across domains.

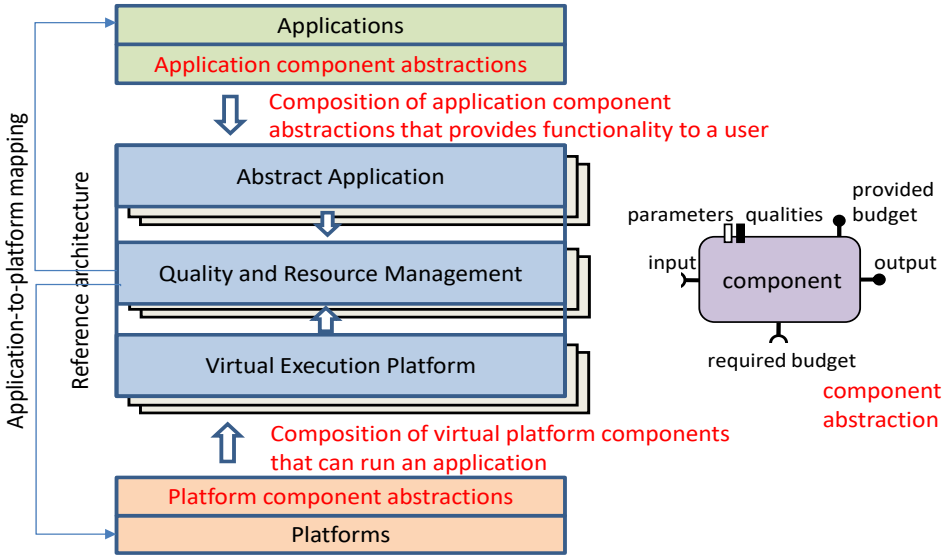

Fig. 3. Technical approach and interfaces – the blue parts form the reference architecture.

IV. FitOptiVis Objectives - Beyond the State of the Art

To achieve the previously presented FitOptiVis objectives (Fig. 2), new technological innovations on SW and HW have to be integrated efficiently to obtain system-level optimizations addressing the full computational chain of the distributed image- and video-processing pipelines. This section presents FitOptiVis foundations, providing an analysis of the state of the art and of the advances contributing to achieve FitOptiVis OBJ1-5.

A. OBJ1 - Reference Architecture and Virtual Platform

State of the art reference architectures exist in the automotive [9] (AUTOSAR) and avionics (*Integrated Modular Avionics*, ARINC-653 [10][11]) domains. They are oriented towards standardizing SW and operating system support interfaces with an emphasis on reliability and certification. AUTOSAR defines a run-time environment that enables dynamic linking of applications. The ARINC-653 standard defines time and space partitioning solutions in real-time operating systems to provide reliable, composable virtual platforms. Compositionality is extremely important for the design of CPSs [12]. FitOptiVis takes the need for compositionality and composability as a starting point and design goal for its reference architecture and for the quality and resource management framework, to achieve the required separation of concerns, between design-time and run-time, among hierarchical levels, and among distributed parts of systems. The reference architecture under development within FitOptiVis has an emphasis on the challenges of CPSs, supports heterogeneity, dynamic environments, and it considers functional as well as non-functional aspects. To date, there is no such reference framework for distributed video and image pipelines within the CPSs domain.

The FitOptiVis reference architecture is built upon state of the art component abstractions [13][14]. Components must provide their operating set points, including functional and non-functional properties, to enable online multi-objective optimization as a basis for *Quality and Resource Management* (QRM). Resource requirements need to be made explicit as virtual platform requirements. State of the art virtualization techniques are commonly used for general purpose computing systems, such as servers in cloud computing infrastructures and stricter forms of isolation among virtual platforms, including non-functional properties, such as performance and energy budgets, are being explored for embedded platforms [15]. Also, there are approaches for combining mobile applications with edge and cloud (ETSI standard *Multi-access Edge Computing* [16]). The coming 5G [17] network will also offer (among others) direct support for using edge-cloud processing (i.e., a possibility for customers to install applications directly to the 5G provider infrastructure) and network virtualization techniques are being developed [18].

FitOptiVis adds communication schemes that offer timing guarantees to enable network virtualization. Moreover, (soft) real-time admission control and set point selection considering application requirements, resource needs and resource availability are required. [19][20]. describe an algorithmic solution for a multi-objective run-time decision making mechanism based on Pareto algebra. This mechanism is abstracted as a multi-dimensional multiple-choice knapsack problem. FitOptiVis is basing its QRM framework on these concepts considering aspects like timing requirements, workload variations, workload redistribution and reconfiguration costs.

FitOptiVis is extending the state of the art by providing a reference architecture for distributed image- and video-pipelines in the CPSs domain that covers non-functional properties and the full edge to cloud infrastructure; component abstractions, resource virtualization techniques, and QRM methods enable real-time multi-objective decision making with respect to functional and non-functional properties, in particular performance and energy. Adaptation and reconfiguration are possible without disrupting quality of service. The proposed reference architecture, illustrated in Fig. 3, enables the fast creation of varying processing infrastructure skeletons that can be verticalized in different silo implementations, by specifying the instantiated components leveraging on design-time support strategies and smart devices which are currently under development. The QRML DSL [4][5] is being developed to facilitate the creation of component models that make the QRM view of applications and platforms explicit. Since CPSs are highly evolvable systems, which may vary their needs and constraints over time, flexible virtual platforms allow abstracting SW and HW details, and they enable the possibility of modifying the processing pipeline at run-time.

B. OBJ2 - Design-Time Support: Model-Driven Development

Today's systems are typically network connected and commonly partly run in a cloud (to access data from anywhere, to offload resource intensive computing, etc.). For CPSs, these cloud-located parts must guarantee (at least soft) real-time properties. The design of an application consisting of multiple distributed edge and mobile devices which transparently cooperate with cloud-located parts and guarantee real-time properties, is an extremely complex task.

Model-Driven Engineering (MDE) approaches are often applied to improve design quality and efficiency. In streaming pipelines for CPSs, a diverse set of modelling paradigms is applied (dataflow, state-based models, agent-based models, differential equations). Moreover, different aspects (functional, performance, timeliness, resource usage) are studied with their own models, thus models of different disciplines need to be brought together [21]. However, mixing models is often not supported in existing methods. Approaches for mixing multiple aspects (paradigms) into a single description (multi-paradigm modelling – MPM [22]) are being developed. Solutions for separate aspects exist, for instance, to guarantee low latencies when processing data in an edge-cloud system (typically with the help of network virtualization [18]). However, there is no approach readily available to combine all the aspects necessary for FitOptiVis into a single solution.

Most MDE approaches that are actively employed today [23] are targeted towards SW systems (on general purpose HW). FitOptiVis also considers HW accelerators and smart devices at the edge that are application- or domain-specific, and not readily supported by MDE methods today. Processing components in streaming pipelines for CPSs are typically heterogeneous, with e.g., FPGA or Graphics Processing Unit (GPU) support, for performance and power efficiency. Such components require dedicated programming and parallelization techniques [24][25][26] that need to be coupled to the MDE. In supporting tools, we observe a raise in abstraction level. For instance, FPGA accelerators can efficiently be programmed in C with high-level synthesis techniques [27][28], while experimental tools allow programming them also through OpenCL [24], taking performance explicitly into account. Other research efforts generate accelerators from dataflow models [29][30][31]. FitOptiVis pursues MDE approaches for HW components and smart heterogeneous devices beyond the current state of the art, in particular including accelerator, GPU, and FPGA support; it copes with the platform diversity seen in existing CPSs and include non-functional properties and the trade-offs between multiple objectives into the process, taking initial research results as starting point [15] [24][29][32][33]. *Design-Space Exploration* (DSE) methods are used to determine (sub)optimal realizations of system components (for instance in [34]). Those solutions are usually static and are optimal only for a particular objective. Design-time methods for DSE need to be extended to allow for reconfigurable components to offer multiple configurations representing different trade-offs, considering the reconfiguration process and reconfiguration cost. Moreover, design solutions for CPSs require awareness of dynamic variation and adaptability.

FitOptiVis is providing integrated DSE, MDE and programming techniques for image- and video-pipelines for CPSs that cope with platform diversity observed in today's practice. Non-functional properties like performance and energy, and trade-offs between these properties are considered. Generation of multiple set points and configurations allows run-time QRM, workload

adaptation and tasks redistribution. The idea behind our approach is to leverage, as much as possible, on implementation-agnostic models to raise the level of abstraction and to cope with a plethora of different domains. Modularity and composability of MDE approaches help in defining optimal-by-construction generic implementations. Design-time DSE, specialization and generation methods (among state of the art and newly designed ones) are adopted to customize the system over the chosen platform (to satisfy its constraints and to optimize the code to be executed) since customization is certainly needed to guarantee efficiency. Tools and automated characterization strategies allow a shorter model to prototype phase.

C. OBJ3 - Run-Time Support: Online Multi-Objective Optimization

Image- and video-pipelines for CPSs are often very dynamic. They may be dynamic for instance in topology, geographic distribution, application workload, quality requirements or resource availability. Static solutions do not suffice. Run-time adaptation, (multi-objective) optimization, and reconfiguration are necessary. Moreover, online monitoring, learning and calibration of quality aspects like performance and energy usage, resource usage, and resource availability are required. Many challenges are ahead on these topics.

Micro-servers with accelerators have been shown to provide opportunity for energy reduction in big data (*Hadoop, Map Reduce*) applications [35][36]. Dynamic workload distribution is well developed for homogeneous general-purpose clouds, but not for heterogeneous platforms seen in CPSs. FitOptiVis is developing transparent integration of computation devices from the edge to the cloud with an emphasis on, possibly distributed, image- and video-pipelines.

Typical run-time systems today are based on static profiles of applications. State of the art run-time monitoring solutions [37] improve on this by providing online feedback about the performance and resource usage of applications. This feedback can compensate for limited accuracy of predictions from design-time models and trigger adaptations or optimizations in the system. Current solutions however are insufficiently customizable and adaptive to fit in the FitOptiVis run-time optimization framework. FitOptiVis is extending existing solutions with a model-driven monitoring approach that can adapt itself to changing conditions, including workload variations and changes in optimization goals.

Online adaptations and optimizations for QRM purposes need to be enforced. The system and components need to be reconfigured. FitOptiVis considers existing *Application Specific Instruction Processors* (ASIPs) and reconfigurable accelerators as part of the computing infrastructure [38]. FitOptiVis is extending model-driven support for reconfigurability within heterogeneous platforms with coarse-grained task-level HW reconfiguration and SW task migration/offloading. Existing reconfiguration solutions may interrupt service quality during such reconfigurations. Solutions have been developed to create virtual platforms and load applications onto them in a predictable and composable way, i.e., in a predictable amount of time and without disturbing the other virtual platforms on the shared infrastructure and the application running on them [39]. FitOptiVis is extending the state of the art by developing methods to reconfigure virtual platforms on a shared physical infrastructure in a predictable and composable way and to redistribute heterogeneous virtual platforms and the video pipelines mapped on these platforms. Together with techniques for run-time multi-objective optimization [20], these reconfiguration techniques form the basis for real-time QRM for image- and video-pipelines for CPSs under development in FitOptiVis.

FitOptiVis provides dynamic workload distribution, transparently integrating heterogeneous computing devices from the edge to the cloud; a model-driven monitoring approach that is adaptive to changing workloads and optimization goals; model-driven support for reconfiguration of accelerator devices; predictable and composable reconfiguration of virtual platforms; and real-time QRM for image- and video-pipelines.

Leveraging on model-based strategies we are going to guarantee the possibility of adapting and redistributing the workload at run-time, by turning on and off given components, by changing the tasks and data partitioning among the cores and the accelerators, by switching among working points, all without disrupting quality of service.

D. OBJ4 - Energy-Efficient, High-Performance, Smart Devices and Components

Efficient realizations of image- and video-pipelines are not possible without energy efficient and high-performance components. For the FitOptiVis domain, such components need not only to be efficient, but they also need to be adaptable, reconfigurable, and amenable to integration into the reference architecture, in order to be seamlessly incorporated in the model-driven monitoring and reconfiguration approach. Moreover, such specialized architectures need to be applied in domains that hitherto mostly apply standard general purpose, hence less efficient, less performant solutions for reasons of ease of development. Energy efficiency is an important topical subject. Specific techniques, such as *Dynamic Voltage and Frequency Scaling* (DFVS),

are being used at different levels during the design flow for both general purpose and application-specific computing resources. FitOptiVis is going to exploit and extend the outcomes of previous successful works on SW, e.g. TTA-based ASIPs [40] and HW, e.g. coarse-grained power management [29], to reach the desired energy efficiency, always considering adaptivity in order to obtain components with several working points in terms of performance and energy, e.g. by exploiting approximate computing to provide energy versus quality/performance trade-offs [41].

CPSs are heterogeneous systems of networked devices. There is a lot of opportunity from processing at the edge of the network, close to the sensors and actuators, to reduce the information that need to be communicated through the network. Smart cameras are an example of such devices for video pipelines [42]. They allow for instance distributed video coding solutions [43] where efficiency gains are obtained by in-network processing and joint coding of multiple video streams. An interesting application for CPSs is image pipelines for extracting object dynamics to act as a sensor in a control loop [44][45][46]. Also virtual reality and augmented reality techniques [47][48][49] are of particular interest for image- and video-pipelines for CPSs. Cost, quality, and performance of all such applications can be improved by smart near-camera processing. FitOptiVis is developing these near-camera processing solutions that integrate into the reference architecture.

Philips FlexVision [50] or *Siemens Large Screen* products provide state of the art display solutions for the integrated rendering of multiple video streams simultaneously. These products cannot interact with their environment and simply try to render the incoming video streams onto the screen. This results in very high bandwidth connections, expensive equipment, and high-power consumption. FitOptiVis is developing smart display solutions by introducing run-time coordination techniques between image sources and displays, and model-based predictive and adaptive system controls.

Smart devices and components for FitOptiVis need to integrate in the reference architecture, i.e., they need to support run-time adaptation across a networked platform, to adapt to available bandwidth or processing resources, and to optimize multiple objectives, including non-functional properties, such as quality, performance, or power dissipation. Current video analytics solutions [51][52][53] do not support such adaptation. FitOptiVis is developing adaptive image processing functionalities, such as object detection or feature extraction, with HW support through accelerators, FPGAs or GPUs [54][46][55].

Dedicated artificial intelligence solutions, related to machine learning, active vision, and face recognition algorithms, are being extended in order to allow for their efficient execution on the edge of the video processing pipelines. Execution efficiency on the edge of processing pipelines may pose hard power and energy constraints. In FitOptiVis we cope with these by porting and customizing those algorithms on dedicated HW accelerators [56][57]. We are extending already existing solutions to support run-time scalability and dynamic trade-offs management among quality, performance, and power (e.g., by means of approximate computing techniques). In particular, the latter contributes to provide flexible and efficient edge processing to withstand the adaptation needs of CPSs.

FitOptiVis looks also at high-performance data-processing platforms such as the TTA [24][58] and rVEX [59]. Current platforms can be reconfigured according to application requirements. However, they are not able to dynamically scale up and down depending on the performance and power requirements for the full range of deployments going from embedded to server applications. This dynamic scalability is under development in FitOptiVis.

FitOptiVis is going to provide energy-efficient and high-performance smart devices and components that enable different reconfigurable working points per component. In this way, they can be effectively employed at the edge of the network and are run-time adaptable and reconfigurable to multiple objectives. Advanced components, such as artificial intelligence-based accelerators, are under development, improving state of the art solutions integrating approximate computing and dynamic trade-off management techniques. The devices and components adhere to and validate the FitOptiVis architecture and provide suitable component models for integral design-time and run-time QRM.

V. FitOptiVis Use Cases

FitOptiVis takes its main requirements, architectures, designs, implementations, demonstration, and validation from 10 different UCs. For each UC, current practices, issues, and challenges are presented in the following, with a special emphasis on the expected achievements and the already presented technical and scientific advances.

A. *UC1 - Water Supply*

The water supply system constitutes a critical distributed infrastructure that requires continuous monitoring, to detect the facility integrity, unauthorized accesses and fast surveillance and maintenance intervention, to restore missing service states.

Potable water represents a primary service of vital importance. Thus, water supply systems require advanced preventive measures and strategies to keep them secure and available to the local communities, and to deter any potential threats or contamination [60][61][62]. For example, the German water supply association decided that all deep wells and water tanks should be equipped with cameras able to transmit the recorded video to a central control office. Monitoring and securing such a critical system, typically spread over a wide area, still represents a challenge. Water pump stations, security access gates, water treatment stations, water reservoirs and pipes should be monitored by means of surveillance devices featuring reliable, scalable, and seamless connection to the control center. The devices should also feature situation-awareness capabilities to guarantee effective management of the water supply system in both normal and critical circumstances. Commonly, real-time monitoring is performed with SCADA systems, not camera equipped, but including different heterogeneous digital sensors (e.g., pressure, water flow, and chemical sensors), and displaying the collected information in a control room. This enables operators to pilot some actuators (e.g., water pump and valves) to control and modify the status of the system according to the data received from the sensors. To detect abnormal system behaviors, it is also possible to compare the real data acquired by sensors with data simulated using specific SW such as Epanet [63], a water pipe simulator. Usage of single or multiple cameras for real-time object and event detection is a well-studied topic [64]. Literature also shows that high- and low-resolution infrared thermography cameras can detect water leaks in pipe systems [65]. In [66], infrared cameras have been installed over quadcopter drones to detect leaks in water pipes.

FitOptiVis design infrastructure, dealing with complete edge to cloud image- and video-processing pipelines, is improving classical SCADA systems by integrating cameras in the processing flow. The traditional SCADA is being turned into an advanced CPSs for multi-source reconfigurable processing accelerating advanced detection algorithms and event-action mechanisms.

B. *UC2 - Virtual Reality*

Virtual reality systems are increasingly important in medical, gaming, and military applications, as well as in the cinema industry. Recent advances in digital photography and video led to the development of advanced 3D vision and display systems. Virtual reality applications should be supported by high-quality video capture, efficient coding and processing technologies and an accurate, fast positioning systems.

Depth estimation can be performed by exploiting six main techniques: time-of-flight (TOF) cameras [67], LIDAR sensors [68], radars [69], structured infrared light projection, learning based single camera algorithms and stereo cameras. Some major challenges affect these depth estimation systems. Some:

- do not provide efficient results when the distance between the object and camera is long [67] or in outdoor applications;
- present a low resolution [67][68] or are poorly performing outdoor [70][71];
- suffer interference when multiple sources are oriented toward the same location [67][70][71];
- present higher cost than CMOS and CCD cameras [67][69];
- are heavy and bulky due to complex scanning HW [69].

Implementing a high-quality and real-time depth estimation system is very challenging using a single camera; these algorithms are computationally intensive and require very large data sets. To compute the depth map for real-time video processing applications, most of the research currently focuses on extracting the disparity information using two or more synchronized images taken from different viewpoints, using CMOS or CCD cameras [72]. The recent development of high-quality free viewpoint synthesis algorithms and their implementations allow realizing glasses-free 3D perception. Although many algorithms are developed for this application, the real-time HW realization of a free viewpoint synthesis for real world images is challenging due to its high computational load and memory bandwidth requirements.

FitOptiVis is overcoming the limitations of multiple sources and outdoor usage. The demonstrator involves a complete multi-source depth estimation system embedding several 3D-based video processing applications (including speed and distance measurement, depth-based image thresholding, tracking, etc.).

C. UC3 - Habit Tracking

Habit Tracking at home is particularly relevant for the elderly population, to assess their physical habits and identify situations where a healthy lifestyle can be supported, and to detect deviations from a standard behavioral pattern, or early discovering of potential deterioration of users' health or wellness.

In current ageing European societies, one of the challenges is to keep elderly people safe and monitored continuously, bearing in mind the protection of their privacy and their personal safety. This need is justified by the reduction of costs of hospitalization (i.e., chronic patients) and the improvement of the comfort and quality of life of citizens from either urban or rural areas [73][74]. In this context, the "tele-alarm" business has emerged. Specifically dealing with *Habit Tracking Systems* (HTS), it is mandatory to enable video processing with real-time performance. In such a context, one can leverage on multi-view (multi-camera) video processing systems that can do robust and efficient tracking of people in different rooms. There are already solutions in the state of the art for people tracking [75][76]. However:

- coping with a massive amount of data and being capable of real-time processing, assessing resource vs. accuracy trade-offs [55],
- handling run-time adaptation through behavioral synthesis [77], and
- ensuring sufficiently robust tracking of people in scenarios with occlusions, or multiple targets

are still open challenges in the current literature.

Some already available solutions are:

- NETATMO indoor security that includes also face-recognition support, but it is not open [78], it does not allow building an application with complete control of the technology;
- *Just Checking* [79], which does not provide activity detection, so it does not support and help users to modify their way of life, by establishing behavioral patterns and making objective performance measurements.

D. UC4 - 3D Industrial Inspection

Industrial inspection applies machine vision to quality control in production processes. Cameras assess the produced objects and automatically determine their quality.

Computer vision extracts information from the real world by analyzing digital images or videos, data can take many forms, i.e., frames, video sequences, views from multiple cameras [80]. Computer vision is often used to produce 3D models from image data, i.e., reconstructing a 3D object from several images. Machine vision, instead, is the process of applying a range of technologies and methods to provide image-based automatic inspection, process control, or robot guidance in industrial applications. It often focuses on applications as vision-based robots and systems for vision-based inspection or measurement [81][82]. The abovementioned disciplines have significant overlap. Computer vision covers the core technology of automated image analysis; while machine vision usually refers to a process of combining image analysis with other methods and technologies to provide automated inspection and robot guidance in industrial applications. In UC4 the focus is on manufacturing applications, where quality control details of final products are being automatically inspected to find defects.

At the state of the art, there are many kinds of computer vision systems; all of them contain these basic elements: power source, image acquisition devices (camera, cables, and connectors), processors, control, and communication network, display and SW. Vision systems for inner spaces, as most industrial ones, contain an illumination system and may be placed in a controlled environment. In inspection context, the following systems are adopted: measurement instruments by MicroStudio [83] and by FARO [84].

FitOptiVis is improving machine-vision-based inspection systems, starting from a novel 3D acquisition system [85][86], capable of capturing objects in an automatic inspection system for quality control in manufacturing.

E. UC5 - Road Traffic Surveillance

Road and railway traffic surveillance, for vehicle detection and recognition, aims at supporting road management, reducing congestions.

Currently, embedded image- and video-processing systems are based on:

- video cameras and specialized HW (sensors and control chips) with focus on efficiency in consumption of resources (price of components) and optimal execution of the most frequent processing tasks needed for pre-processing, conversion of color models, debayerisation, etc. [42],

- computers, standard or embedded ones with camera interfaces. These systems are often high-performance and built/programmed in an optimal way in terms of price and performance of the applications or their parts.

The above-mentioned solutions allow building efficient systems, but very few products and related technologies adopt cameras with integrated processing power, which would allow for more efficient local edge processing.

The ability of the edge processing enabled/simplified by FitOptiVis is going to benefit the involved applications from two points of view: i) it reduces the computational load in the server and communication load of the interconnecting network, thus lowering the operational costs of the whole system; ii) the technology is an enabling factor for those (possibly new) applications where the communication and/or the computational load of the server systems is at the limit.

FitOptiVis allows the composition and run-time management of heterogeneous stream processing pipelines, where edge components can be enforced with programmable HW capabilities (FPGA and CPU), easy programmability (ideally compatible with PC, at least to some extent), and security features.

F. UC6 - Multi-Source Streaming Composition

UC6 involves the definition of an embedded high-performance video compositor capable of rendering eight simultaneous streams on a screen. Tight coordination among the video sources, and with the compositor is required, as well as adaptivity to variations in setting. If the screen layout changes, the compositor needs to dynamically adapt.

At the state of the art, video streams are simply scaled to the right size and composited onto a screen. Scaling algorithms are currently adapted scaling medical information, which means that graphics are distorted (especially when scaling down) and there is not management of the video sources themselves neither can the image sources interact.

Currently adopted systems are:

- *Philips FlexVision* [50] and *FlexSpo*t equipment: up to 8 DVI video streams out of 16 are simultaneously composited onto a single screen;
- *Siemens Artis* [87] system: up to 21 digital video streams are composited onto the screen;
- *Barco Nexxis* [88] display system: one of the first display systems which uses Ethernet-based video transport in an operating room environment. It can display up to 4 streams on a monitor.

These products are currently the most adopted image display systems in a clinical operating environment. They cannot interact with the environment; therefore, what they do is try to render the incoming video streams onto the screen. This results in very high-bandwidth connections, expensive equipment, and high-power consumption.

G. UC7 - Sustainable Safe MRI

To facilitate *Magnetic Resonance Imaging* (MRI) use in long interventional procedures, in pediatric imaging, and in the emergency setting, a low energy deposition method for 4D imaging is intended to be developed. Goals are to reduce energy consumption and provide thermal losses models to predict, control, and minimize thermal load, while maintaining superior image quality.

MRI equipment requires both actuators (creating electromagnetic fields) and sensors (capturing electromagnetic fields) to create non-invasive images of the human body without using ionizing radiation. Timing of the creation and capturing of these fields is critical because phase differences are the underlying basis of imaging. To meet the low latency requirements and the strict control of power deposition in the patient, typical MRI architecture consisted of a central digital controller connected to distributed analogue controllers (for both actuators and sensors). Following the advances in CPSs, MRI is adopting decentralized control [89] and model-driven design [90].

In EMC^2 [91], post-processing viewing directly on the MRI console was investigated such that image quality can be assessed more easily by the operator, leading to less patient recalls [92]. With the state of the art technology, it is not yet possible to run reconstructed advanced scans on a centralized resource, and virtualization does not meet latency requirements.

In FitOptiVis, MRI systems is being optimized by developing and applying technology to balance processing on edge devices and central processing, while safeguarding the latency, safety (i.e., thermal load) and image quality aspects. To this purpose the FitOptiVis architecture is being integrated. Design- time models are under creation and abstraction in run-time models for efficient execution, thereby minimizing thermal load, while providing superior image quality.

FitOptiVis design methodologies allow for power- and performance-aware system customization that can improve currently available distributed digital scanners.

H. UC8 - Robots Calibration

Robotic arms or manipulators require calibration to be done periodically or after each geometry change. This UC is meant to provide fast and automated calibration, getting rid of any mechanically connected measurement device.

Calibration is currently a complex process involving mechanical connection of a measurement device to the robot being calibrated. Once calibration is done, there is a necessity to move the measurement system from one calibrated robot to another. For big robots it means moving equipment of considerable size. On the other hand, for small robots the mechanic of the measurement device may influence the position of the measured parts (as the measurement force is not zero) and thus influence the calibration results.

Depending on the demand factors of the application and available budget, various calibration solutions across the price spectrum are used (mechanical 3D scanners/calibrators, laser trackers or vision controllers) and, unfortunately, due to the high price of the calibration equipment, the calibration is in some cases omitted at all, causing decreased quality of the products. A few calibration solutions, as those proposed by Hexagon [93] or by Denso [94], are available. They leverage on computer vision, but their price excludes them from being widely used.

The FitOptiVis UC8 aims at deploying an easy-to-use calibration solution, based on computer vision, where the extremely high setup costs are avoided and the most appropriate solution (given the system) to be inspected is chosen.

I. UC9 - Surveillance of Smart-Grid Critical Infrastructure

UC9 focuses on active surveillance for the prevention of potential harm to EU smart-grid critical infrastructures. The detrimental effects caused by disruptions on the society and citizens must be minimized. This requires a hybrid network that combines information from video surveillance with critical control information.

The utilization of smart surveillance systems that include drones is spreading (as already seen for UC1) and they are also taking over additional tasks such as monitoring and inspections [95][96][97]. Several challenges must be tackled in surveillance systems:

- they are usually isolated and spread in remote areas, posing in conventional constrained networks problems both in deployment and in communication with the cloud;
- protection of critical sensitive data must be ensured, along with the reduction of the security crew operations cost;
- efficient and real-time processing of video streams to promptly trigger alarms, while reducing false alarms.

Nowadays, in the most recent systems, multi-view surveillance is possible, but autonomous operation of the surveillance platform, to reduce costs and to allow for local processing, is still an open problem.

Ensuring bounded latencies for the communications and the capacity of reserving part of the bandwidth in a conventional network for priority traffic transmission is one of the key elements of this UC, performed by using deterministic networks with time sensitive networking [98] and *High-availability Seamless Redundancy* (HSR), based on the standard "IEC 62439-3" that offers seamless failover against failure, and higher performance and synchronization.

The system proposed in UC9 deals with a hybrid approach that integrates active surveillance and passive traditional video processing. New high-performance smart components for image processing pipelines estimating motion and depth cues [54], independently moving objects (meaning objects moving willingly), or background-foreground subtraction [55] are under development since current off-line computation cannot deal with real-time performance. Smart adaptation through active vision is still not described in the literature [99]. As stated above, the use of drones to monitor critical infrastructures is being dramatically increased. They can help assessing a potential danger or they can provide additional information to take smart decisions. Amongst the current solutions one can find:

- *AirRobotics* that targets inspection in industrial environments [100];
- *Airborne Drones* that targets illegal poaching in protected environments, or traffic monitoring [101].

On the other hand, the smart grid implies digital data, Ethernet communications, operational technology systems and interconnection with information technology systems such as: security systems, maintenance and supervision systems, geographic information systems that use image- and video-data and increase the performance required. Sensors, actuators, and other smart devices such as *Remote Terminal Units* (RTU) are interconnected, sending and receiving the information from the control center in real-time. In this context, the standard "IEC 61850" is the new reference for the communication networks and systems in the electrical substation, combining the convenience of Ethernet with the performance and security which will be

essential in the digital substation of the future. This standard includes HSR that offers seamless failover against failure and higher performance and synchronization. The UC addresses the implementation of HSR in control devices (RTU), improving the communications in the smart grid with respect to performance, availability, and synchronization.

FitOptiVis technologies in this context allow offering improved multi-source and multi-sensors processing capabilities.

J. UC10 - Autonomous Exploration

Various challenges have to be faced by a robot system while conceiving with planetary or hostile environment exploration [102][103]. Autonomous Exploration UC involves the definition of the next generation video processors that will constitute the core of the future earth observation and robotic planetary exploration missions. The aim is to make the processing adaptable to different non-functional critical parameters (i.e., available power or connection bandwidth) and unexpected functionalities under degraded conditions, or failures of some of the system elements due to the challenging environment.

Reconfigurable video data processing and sensor data acquisition systems for space applications are currently introducing technologies quite new to the space market as smart sensors, multi-core processors, or *Systems on Chip* (SoCs). In the space business, these technologies are newcomers that are entering the market at an extremely slow speed, especially when compared with the promised advantages that such systems may bring in terms of performance improvement. FitOptiVis is going to speed up the entrance of modular video processors by tackling the criticality of the space borne systems and the associated validation and certification procedures through modular approaches that allow gradual deployment, reuse and validation of the algorithms and associated data processing devices. There is a lack of methodologies and tools to support the exploitation of these new technologies in the scope of systems, which are compliant to the strict requirements of power consumption, performance under critical conditions, safety, timeliness, security, and reliability peculiar to the space applications.

FitOptiVis is proving the validity of the different architectures for space application opening, in turn, new application domains to the use of reconfigurable video processors and dynamic video processing algorithms. This UC targets a final product application and, therefore, it must be guaranteed not only compliance with the functional requirements, but also, to non-functional requirements currently specific to space applications.

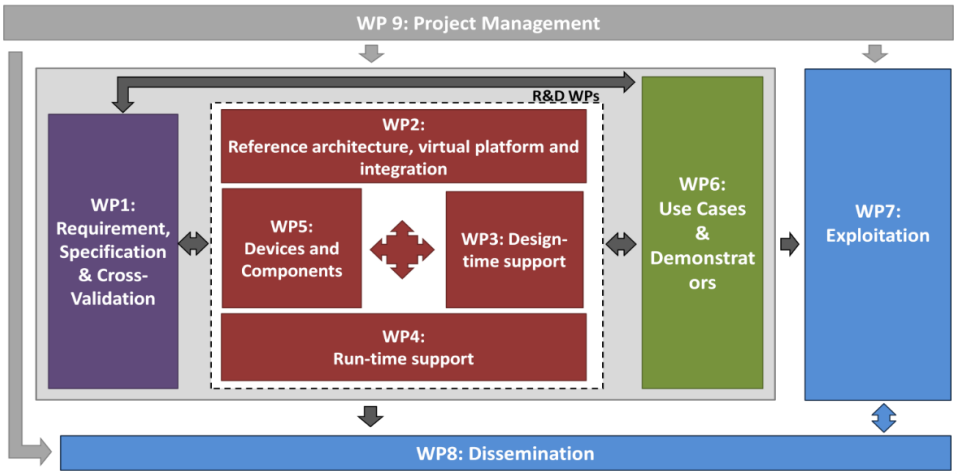

Fig. 4. FitOptiVis Work Plan.

VI. FitOptiVis Implementation

The mission of the FitOptiVis implementation work plan (Fig. 4) is (i) to achieve technical objectives of the proposal, (ii) to maximize collaboration among partners and (iii) to guarantee a solid impact of the project. We aim to have impact in terms of advancements with respect to the state of the art, as well as improvements with respect to current industrial best practices, and we intend to foster a concrete market uptake of the project outcomes.

FitOptiVis technical activities are organized as follow:

- *Requirements, specification and cross-validation of the results (*WP1) – We apply two sources for requirements: the UCs and the industrial/academic knowledge on the state of the art and current issues. Validation, both partial and final, takes results from demonstrators on the UCs and the evaluation against the requirements, its outcome is a report with the KPI metrics.
- *Reference architecture, virtual platform and integration* (WP2) – The reference architecture provides component abstractions for the image- and video-pipelines and their implementation platforms, as well as platform virtualization techniques, including the QRML DSL. They form the basis for QRM for image- and video-pipelines, in terms of budgets for resources like frame rates, power dissipation, required processing, communication and storage. Emphasis is on technology supporting multi-objective optimization for, at least, performance and energy. System-level concerns like distribution and (re-)configuration are addressed. The architecture needs to make sure that the image -and video-pipelines and the run-time support can work on a heterogeneous network of HW devices.
- *Design-time support* (WP3) – The definition of a model-based working methodology involving methods and tools for predicting, simulating, and estimating at design-time resource usage is the core of design-time support. In addition, methods, SW libraries, reference designs, HW/SW co-design [104] and compilation techniques are intended to improve the resource behavior of the final system, considering the heterogeneous and changing structure and resource needs of the final system. The development concentrates on all video/image processing building blocks and the run-time support, including HW IP/accelerators, SW applications and sensors.
- *Run-time support* (WP4) – All the technologies that are intended to implement real-time resource management within the system constitute the run-time support of FitOptiVis. It is delivering components considering both the actual implementation that run on the final product and the models to integrate in the system abstraction. It involves monitoring, measuring components [8], and control components implementing the algorithms.
- *Devices and components* (WP5) – We are developing and selecting HW and SW devices, components and configurations that are best suitable for optimal energy and performance use. For each component in the FitOptiVis component library a compliant model view will be available at the end of the project, which may present different levels of abstraction, depending on its usage. Components may have different configurations, and being able to support different trade-offs, which could be exploited by WP4 technologies for run-time dynamic support and reconfiguration.

VII. Towards the end of FitOptiVis

The following provides a summary of the main technical results obtained in FitOptiVis in the first 30 months.

A. WP2 - Reference architecture, virtual platform and integration

Along the project progress, the development of the reference architecture has been consolidated. It ties together the following elements:

- a conceptual architecture for QRM of image- and video-pipelines (see Fig. 3);
- a component interface model that captures the component exterior properties that are relevant for QRM, as well as different types of compositions of such components into larger components;
- a mathematical semantics that defines component interfaces, compositions, constraints, and optimization criteria in terms of a mathematical constraint formulation [6];
- a DSL, called QRML, to specify component models [4];
- a visualization framework that connects the system model to the FIVIS monitoring framework [105];
- virtualization techniques to effectively, dynamically manage shared resources through virtual platforms;
- a QRM reference framework (refining the middle blue box of Fig. 3).

This reference architecture is under evaluation through the project UCs.

The *reference architecture* is visualized in Fig. 3. It distinguishes applications and platforms and captures the mapping of application tasks onto platform resources with an emphasis on application-quality and platform-resource management. It also explicitly captures (re)configuration options of components and QRM goals. The sharing of resources by multiple applications

in a predictable and efficient manner is supported by virtualization techniques that support the partitioning of resources into individual budgets that form virtual platforms that can host an application. The challenge behind this is being able to express the budget in an independent, predictable and composable form, to ensure that the utilization is sufficiently high and that (models of) the resource budgets are simple enough to be effectively used for QRM. An important aspect of the project research is to establish such combinations of budget models and resource management strategies.

We have developed a *component interface model* that can coherently describe platform and application components and their interfaces. Compared to existing component and interface models, we emphasize aspects of QRM [4]. From this point of view, the characteristic aspects of a component interface are the qualities it offers (in our terminology, qualities cover also 'negative qualities', costs, like power dissipation or required resources from a reconfigurable fabric), the (relevant properties of) functional inputs and outputs that form the image or video pipeline it offers, the provided (typically for a platform component) and/or required (typically for an application component) resource budgets, and its configuration parameters (see Fig. 3). Components explicitly can have multiple configurations and the interface properties may vary with the configuration or set point of the component. The component model further describes how components can be composed and how the interface of the composition relates to the interfaces of the components from which it is constructed. Constraints can further be expressed to govern the possible compositions and the valid system configurations. Fig. 5 shows an example component model in the QRML DSL. `component FaceDetection`, for example, describes an application component that requires two budgets, `ImageCapturing` and `ImageAnalysis`, and its operation is captured with quality latency (`lat`). A constraint (`lat.latency==imgAna.latency`) captures how its latency depends on the budget it is provided with. Composition of components is illustrated with the `component SmartCamera`, which consists of two sub-components.

```
component FaceDetection {
    outputs Image out;
    quality Latency lat;
    lat.latency==imgAna.latency;
    requires ImageCapturing imgCap;
    requires ImageAnalysis imgAna;
}
…
component SmartCamera {
    alternatives {
       component SmartCameraNormalMode scnormal;
       component SmartCameraAdvancedMode scadvanced;
    }
}
component SmartCameraNormalMode {
    supports ImageCapturing imgCap;
    supports ImageAnalysis imgAna { latency==25; };
}
component SmartCameraAdvancedMode {
    supports ImageCapturing imgCap;
    supports ImageAnalysis imgAna { latency==50; };
    supports FaceIdentification faceId
       { qual==low; latency==20; };
}
…
typedef Latency {
integer latency;
}
typedef Quality {
enumeration {low, medium, high} qual;
}
```

Fig. 5. A QRML snippet of a Biometric Access Control System

The component model is further equipped with a *precise mathematical semantics* that defines the concepts in the model [6]. Considering the multi-objective nature of the QRM problem, the semantics is given in terms of Pareto algebra [19]. The semantics relates a component model to a multi-objective constraint optimization problem. We have developed tool support to automatically derive such a constraint problem from a model and express it in terms of the input language of the Z3 constraint-

solving tool [106], to find optimal solutions [4]. To support the development of component models a *domain-specific language* has been defined, called *QRML* (pronounced as *caramel*) [5]. The syntax is being developed to allow for a convenient specification of a model by system developers and to support the definition of the interfaces that are important for QRM. Fig. 5 shows an example. Initial feedback from users is used to improve language for a next version. The language is supported by tools. An Eclipse plugin can be generated that allows editing of QRML models, model validation and the generation of some derived artifacts, such as a visualization of the component structure. In addition to the Eclipse-based tools, to enable low threshold access to the language and tools, a web-based tool has been developed that is embedded in the https://qrml.org web site. The website incorporates a model database and provides the visualization and code generation tools that are also available in the Eclipse plugin. The language and tool infrastructure are developed with the explicit intention to enable easy refinement of the language and associated tool towards specific application domains. Domain-specific concepts can be embedded in the language and exploited, for instance, for domain-specific code generation. This approach has been used in the project for the development and code generation targeting UML/MARTE models and tools to, in addition to the standard QRM related concepts, specify component services and to allow automated generation of C++ code templates for monitoring of properties at run-time. This specialized language, called SDSL, and its code generation features are under integration into the https://qrml.org web site. We are also investigating a domain-specific extension for timed dataflow models integrating a compositional semantics of timed dataflow models and an analysis model for dataflow graphs that relates their performance to specific abstract processing budget models.

The project is also integrating *visualization techniques* for the component model in the FIVIS visualization framework [105]. FIVIS is also used in WP4 for run-time monitoring of data from the system in operation. This integration allows the run-time data about qualities and resource usage to be intuitively linked to the conceptual component model of the system, to see the configurations that are active, to validate the accuracy of the models. To this end, a QRML visualization framework has been developed, called QRMLVis [4][107] as a react component that can be easily integrated in web-based applications such as the FIVIS platform and web site[1].

We are investigating *virtualization techniques* in the context of architecture models and architecture description languages in UML-MARTE, run-time support for dataflow models, and in the CompSoC [108] composable architecture. A QRM framework has been developed, based on the reference architecture, and implemented as a prototype demonstrator which virtual platforms can be dynamically created, removed, and modified in a predictable and composable way [109]. These virtual resources are described by abstract and sufficiently simple budget models that are effectively exploited for online QRM. The framework also allows budget models to be combined at different levels of abstraction, for instance, memory budget defined as a total number of bytes, or as a specific memory address range. The methods of implementing virtualization in HW and/or SW themselves are subjects of WP4.

The models allow us to define system models for QRM. We have refined the conceptual architecture of Fig. 3 with a *reference architecture for a QRM infrastructure* that exploits the models to effectively determine optimal configurations and reconfiguration for a system and initiate reconfigurations when needed. We have identified the components that play a role in this architecture, such as brokers, resource managers, application quality managers, etc. An instance of the reference QRM architecture has been developed for the context of the CompSoC platform. We intend to verify its suitability also for other platforms in the future.

In an ongoing effort, we are modelling UCs in the FitOptiVis project, or rather components belonging to UCs, and with the QRML language, to demonstrate how the UCs relate to the overall architecture and to use the feedback from the UCs and from the modelling effort to further refine and improve our models and the QRML language.

B. WP3 - Design-time support

Different methods and tools have been adopted, developed, and extended for updates and releases of design-time resource related technologies.

Supported tools and design flows can be classified considering the main pursued target (even if some of them are transversal to different classes).

[1] https://qrml.org

Model-driven engineering for qualities optimization (MDEQ): this class groups tools in charge of estimating and optimizing resources and qualities, such as energy or performance, leveraging on model-driven design. The main MDEQ methods/tools are:

- S3D: a modelling framework for real-time video processing systems distributed from the cloud to the edge, available as free for research. S3D main features are model capture (based on UML-MARTE), performance analysis and SW code synthesis [110].
- HEPSYCODE: an open-source toolchain [111] driving the designer from specification to implementation of heterogeneous parallel dedicated systems. It uses Eclipse MDE technologies, a customized SystemC simulator and an evolutionary genetic algorithm for architecture definition, HW/SW partitioning, and mapping activities.
- SAGE-VS: an open-source suite of tools [112] for system formal verification. One of its main features is checking model consistency according to system requirements which, being variable at run-time, need a complex analysis to ensure that they can be always met [113].
- IMACS: a framework for closed loop systems with image or data intensive processing. It supports design (MatLab front-end for control and processing), analysis and validation (simulations including physics camera/sensors) and code generation for multi-core platforms [114].

Programming and parallelization support (PPS): this class groups main FitOptiVis methods/tools that offer to users support for the design and programming of customized processors, as well as for the implementation of embedded systems. Two main methods/tools belong to PPS class:

- TCE: an open application-specific instruction-set toolset for the design and customization of processors, based on the energy efficient Transport Triggered Architecture (TTA). TCE provides a complete retargetable co-design flow from high-level language programs to synthesizable processor RTL and parallel program binaries [115].
- RIE: a methodology and C++ library for component-based implementation of embedded systems [116]. RIE provides support for run-time re-configuration of SW components that could have several implementations that are selected at run-time.

Acceleration support (ACCS): this class groups the main FitOptiVis tools that offer support for the design and deployment of HW accelerators, including high-level programs to HW tools as well as HW generators oriented to particular applications or specific acceleration techniques. The main tools that belong to ACCS class are:

- MDC: an open-source automated dataflow-to-HW tool for the generation, system integration and management of coarse-grained reconfigurable accelerators [117][118]. The tool is completely domain- and algorithm-agnostic, and has been applied in several different contexts, from video-coding [119] to more signal processing applications [120].
- NeuDNN: a configurable C/C++ library, providing the APIs to seamlessly execute the CNN in SW or accelerated with the NEURAghe solution [121].
- DTRC: an automated dataflow-to-HW tool for configuration and system integration of design-time resources for Zynq and Zynq Ultrascale+ systems with Debian OS and HW accelerators [122].
- DTRiMC: a tool for system integration of IPs designed, modelled, and validated in Xilinx Model Composer and Xilinx System Generator for DSP [123]. It supports two reconfigurable accelerators with 8 channel Single Instruction Multiple Data datapaths (8xSIMD) which, in comparison to ARM Cortex A53 running at 1.2 GHz, accelerates floating point matrix multiplication by 5x.

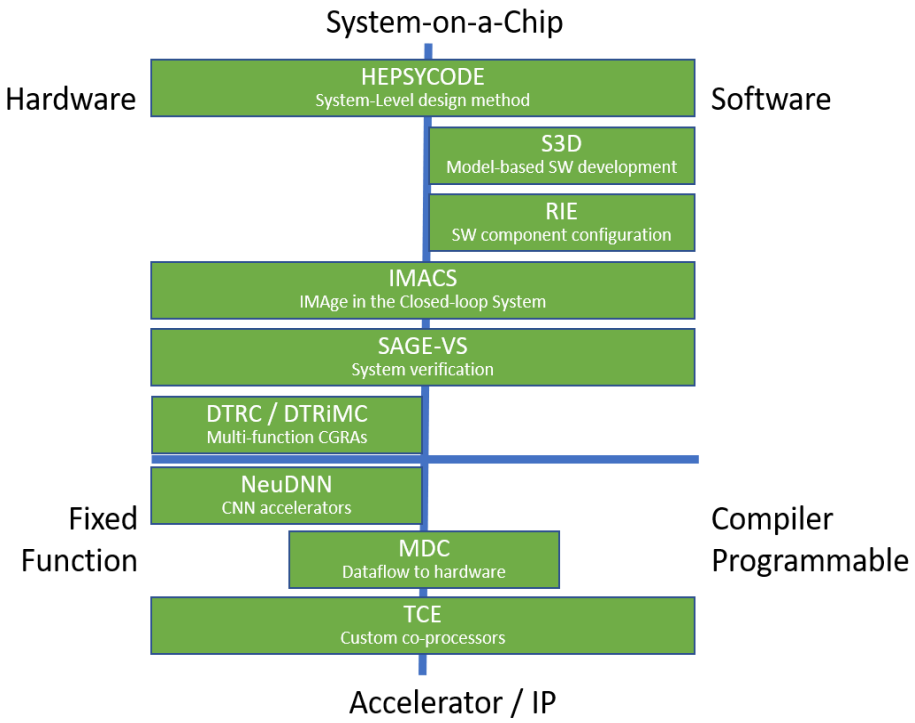

Fig. 6. Tools and design flows developed in WP3.

Supported tools and design flows are summarized in Fig. 6, where further classifications are also highlighted. In terms of targeted device, few tools, such as MDC, and TCE, address single IPs or accelerators, while others address entire SoCs. Most of the tools space from HW to SW, taking care of both worlds, while others are focused only on HW, such as MDC, NeuDNN, DTRC, DTRiMC, or on SW, such as RIE and S3D. Lastly, in terms of flexibility, there are tools supporting compiler programmability only (RIE and S3D), others supporting fixed function devices (NeuDNN, DTRC and DTRiMC), others placing in the middle thanks to reconfigurable computing (MDC), and others being transversal with respect to the configurability of the target platform.

All the presented tools/methods classes are meant to be assessed over different UCs. TABLE I. is presenting the WP3 technologies versus UCs coverage matrix. Please notice that, the full coverage of UCs with WP3 technologies is not going to be achieved, but this was already known from the same project objectives.

TABLE I. WP3 TOOLS ADOPTION IN UCS.

	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10
MDEQ	x									
PPS		x				x				x
ACCS	x							x		x

Improvements, extensions, and advancements on the described design-time tools/methods are expected also for the final part of the project.

C. WP4 - Run-time support

The provided solutions for run-time management of applications have been implemented with reference to MAPE-k loop, which generalizes the concept of a control loop (Ctr-L) for adaptive systems. Such Ctr-Ls, described in the following, can be nested to form a hierarchy of Ctr-Ls operating at different time scales:

- *Managed-Latency Edge-Cloud Environment* (MLEC) - The goal of this Ctr-L, described in Fig. 7, is to provide soft real-time guarantees to applications executed on top of a virtual environment. A developer is required to provide the application and its soft real-time requirements. The platform then monitors the state of the cloud and the performance of applications deployed in the cloud and attempts to maintain a deployment configuration that satisfies the declared soft real-time requirements, as shown in Fig. 7. New applications are admitted into the system only if their performance and the performance of the already-deployed ones can be guaranteed. In both cases, the system uses an algorithm for predicting the upper bound of response time of an application when sharing HW resources with other applications [124]. This allows the system scheduler to evaluate different scenarios, either in attempt to optimize resource usage, or to find configurations that are more likely to satisfy the performance requirements. The MLEC framework serves as an execution platform for periodic and data-triggered analysis tasks defined in FIVIS to ensure that these tasks provide timely results in UCs which include FIVIS in the top-level control loop.
- *CompSoC platform* - The goal of this Ctr-L, described in Fig. 8, is dynamically managing quality and resources of composable and analyzable hard real-time applications running on a system-on-a-chip, which is abstracted through Virtual Execution Platforms (VEPs). The Ctr-L is performed by a predictable and composable deployment framework that exploits application and resource models described with the component interface models developed in WP2. The framework is equipped with a Pareto optimization engine that plans reconfigurations at run-time (including adding, removing, and/or modifying components and/or their compositions) such that they have no interference with the operation of existing applications, and they are done within a time bound. Currently, adding/removing applications and VEPs have been developed, and other reconfiguration scenarios are under development.
- *Reconfiguration in Processor-CoProcessor systems* (RPCP) - The goal of this Ctr-L, described in Fig. 9, is to deal with run-time adaptability at HW level [125][126], specifically referring to multi-purpose co-processing units. A developer is required to describe its custom accelerators with a tool named MDC [118][119], based on dataflow models. The Ctr-L exploits coarse-grained functional and non-functional reconfiguration.

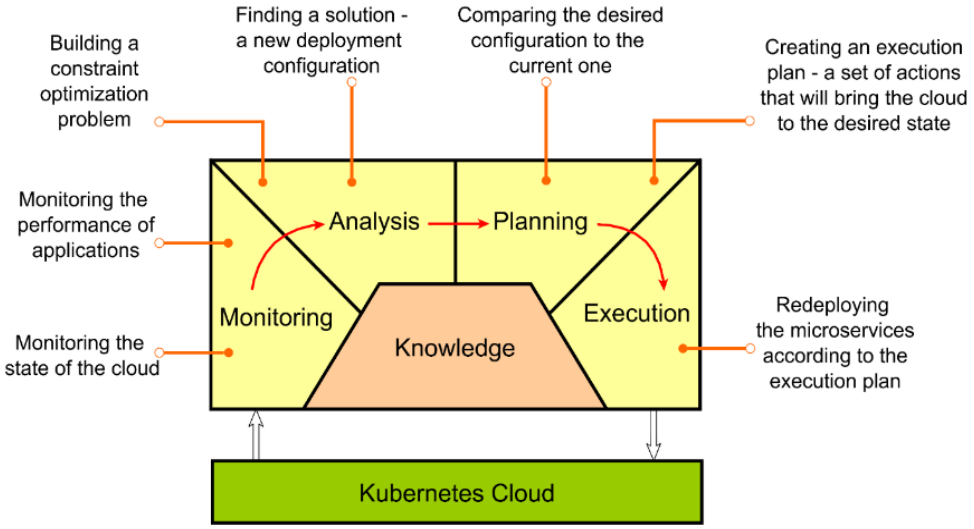

Fig. 7. Ctr-L for reconfiguration in MLEC

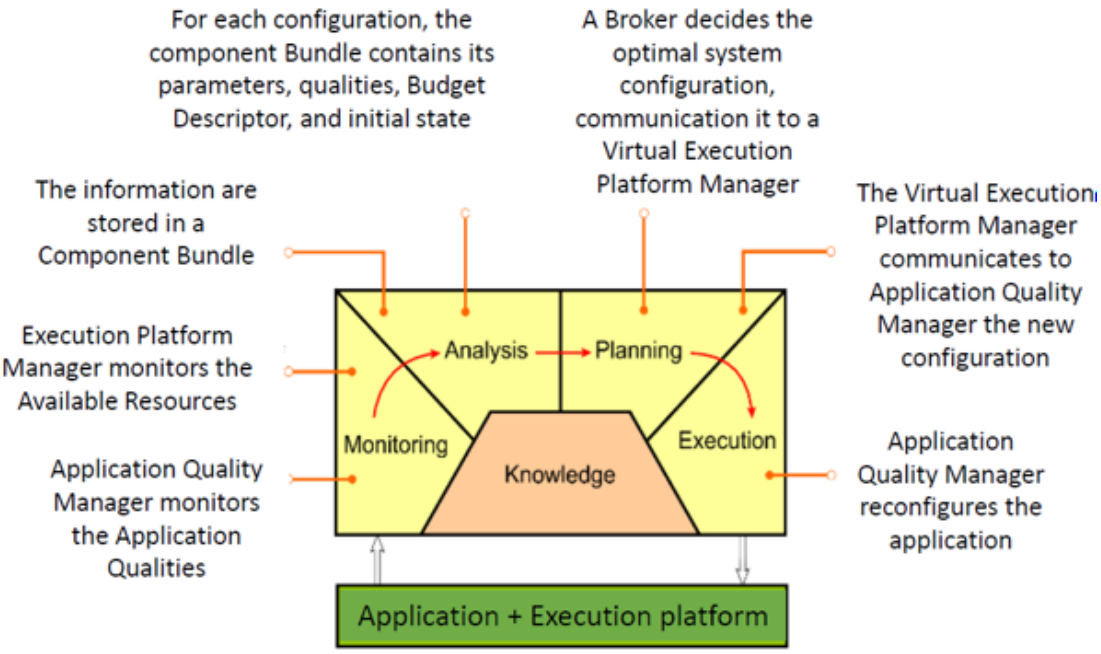

Fig. 8. Ctr-L for reconfiguration in CompSoC.

Similar Ctr-Ls for adaptive systems have already been explored by other studies at the state of the art and projects, as in [127]. What is new here is the assessment at various level of abstraction and the exploitation, in most of the cases, of the *Virtual Reference Platform* developed in WP2 to take run-time decisions upon adaptation and QRM.

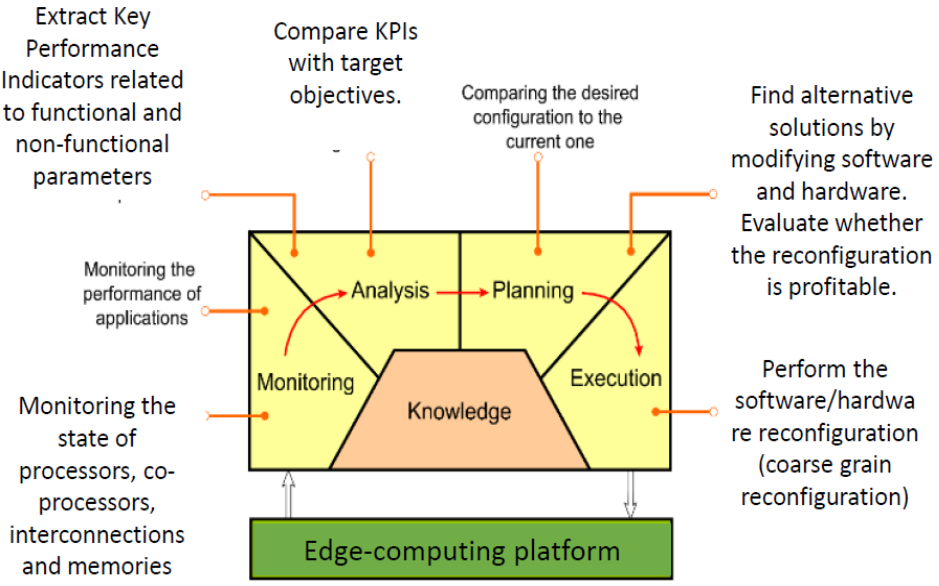

Fig. 9. Ctr-L for RPCP.

For monitoring, AIPHS [8][128], a framework to build HW monitoring systems with controllable intrusiveness, has been developed. The framework can monitor processors, coprocessors, memories, and buses, with the possibility to target multiple monitoring purposes. For example, supposing to have two requirements, namely MON1 and MON2, that ask for a monitoring process on the final target in order to provide run-time verification and a performance evaluation, the framework allows to

build a HW monitoring system that satisfies both MON1 and MON2 with an impact on occupied area that is extremely low compared with state of the art commercial monitoring solutions [129]. For the HW reconfiguration, investigation on the effective Dynamic Partial Reconfiguration (DPR) on hard real-time systems has been performed, with the final project goal to evaluate, at run-time and using the generated HW monitoring systems, when is profitable and feasible to perform multiple DPR [130].

The support for development of resource managers is enforced with the provision of standardized programming APIs to harness all available resources. To this end, a distributed heterogeneous device run-time SW stack based on the heterogeneous parallel programming API OpenCL was developed up to a prototype level. The run-time environment called PoCL-Remote (PoCL-R) was added to the open-source OpenCL implementation framework PoCL [131] as a driver. Adding a distribution driver completes the platform since it can now provide access to all compute resources, both local and remote, from fixed function HW accelerators to remote GPUs controlled with proprietary drivers. This work enabled the possibility to further adapt a high-level programming model, namely OpenMP [132], employing the developed stack, as illustrated in Fig. 10. Finally, different platform templates with easy-to-use implementations and support to prototyping, tailored for heterogeneous platforms (e.g., Xilinx Zynq), have been provided.

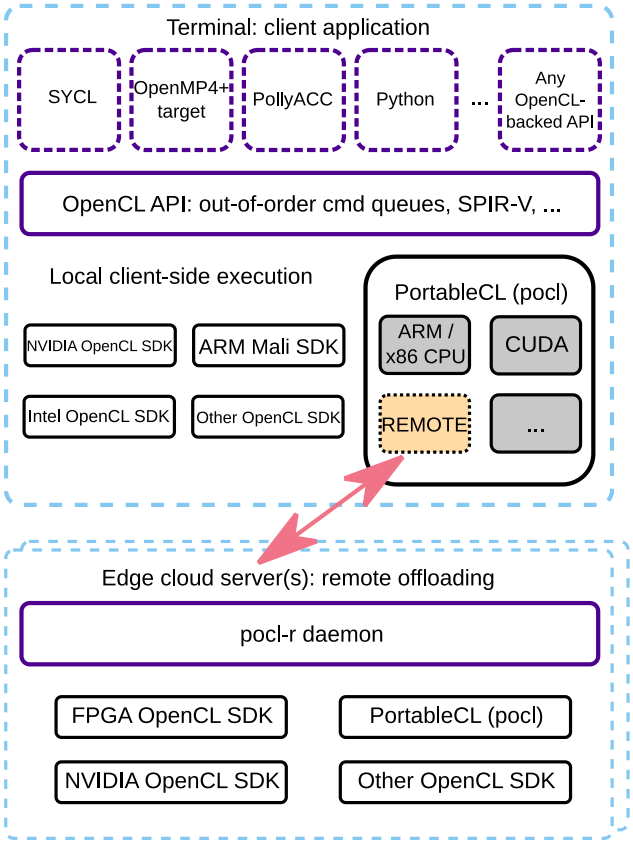

Fig. 10. Overview of the SW stack for an application using PoCL-R. The OpenCL API can be used directly for maximum efficiency, but also as a middleware for improved productivity APIs on top of it.

The reported solutions are still under development, and the ten UCs are currently starting to make use of them to satisfy requirements and to further elicit improvements of the same FitOptiVis solutions. For instance, in UC2 a demonstrator is being constructed where a mobile phone-based augmented reality application is partially rendered utilizing near-by cloud-edge GPU resources. Results have shown enhancements of object-rendering and frame rate. Another interesting example is given for UC3, where PoCL-R is used to seamlessly offload motion estimation computation to the server, as if this server were a local resource, thus reducing the processing time [133]. During the last part of the project, we expect to refine and complete all the described

technologies according to the feedback gathered from the UCs. TABLE II. is presenting the technologies versus UCs coverage matrix. As for WP3, the full coverage of UCs with WP4 technologies will not be reached, but this is known since the beginning of the project and is discussed in the project objectives.

TABLE II. WP4 TECHNOLOGIES ADOPTION IN UCS.

	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC10
MLEC			x	x						
COMPSOC						x				
RPCP	x									
POCL		x	x							
PTL								x		

D. WP5 - Devices and components

For the purposes described in the previous section, the WP is carrying out several parallel activities. The first one is devoted to the definition and description of both i) state of the art commercial components and algorithms, capable of addressing the needs of FitOptiVis UCs, and ii) novel SW (new algorithms and SW IPs, i.e., next generation distributed digital and SW components for MRI control and data processing) and HW (sensing and actuating components, i.e., smart industrial cameras; enhanced image acquisition devices, with on board processing capabilities) IPs. Since in the context of FitOptiVis, dealing with edge to cloud solutions, the exploitation of low-power and low-energy HW for high-performance is of paramount importance, the second activity WP5 is carrying out is related to accelerators. Novel co-processing elements, to be used as a part of the heterogeneous processing platforms, are developed within FitOptiVis: i.e., an 8xSIMD floating point accelerator, a license plate detector, coarse-grained application-specific reconfigurable accelerators. The activities of WP5 are also focused on the definition of proper communication mechanisms for edge-to-edge and edge-to-cloud processing pipelines, allowing local pre-processing solutions for minimizing communication load and enabling fast point to point local interactions. Edge distributed processing is also exploited for cutting down time-to-decision and improving security and privacy.

All components proposed in WP5 are bringing IPs innovations that can be classified as hardware (HW), software (SW) and communication (COMM):

- *HW IPs innovations*: image- and video-pipelines (acquisition, composition, transmission, rendering), image- and video-detection and classification (features extraction, neural network-based classification), MRI application.
- *SW IPs innovations*: coding engines and image processing libraries, classifiers (as behavioral analysis or face recognition systems), remote/distributed monitoring (like person tracking or personal assistant).
- *COMM IPs innovations*: low latency network communication, real-time distributed control systems, communication monitoring support, seamless big data communication, heterogeneous traffic support, secure embedded system-of-systems environment.

TABLE III. WP5 TECHNOLOGIES ADOPTION IN UCS.

	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC9	UC10
HW	x	x		x	x	x	x	x		x
SW	x	x	x					x	x	x
COMM	x	x	x					x	x	

All the previously itemized solutions are exploited in many UCs. TABLE III. is presenting the technologies versus UCs coverage matrix. In summary, the current composition of FitOptiVis components library is the following: 11 processing and acceleration components (HW); 9 SW components (SW); 4 communication-oriented components (COMM); 2 miscellaneous sensor/actuators/composed components (MISC). The progress of the 26 WP5 components is summarized in TABLE IV. where they are listed according to the defined categories, and they are labeled with IDs defined in deliverable D5.1 [134]. The main activities which are currently performed on them (development [Develop.], refinement [Refin.], testing [Testing] and integration

[Integr.]) are highlighted. Please note that Refin. means either extending or optimizing an already developed component, while Integr. means either integrating the component with the work made on other WPs (e.g., models in WP2 or tools in WP3) and/or integrating it within the UCs. As visible from the table, most of HW components are still under development, even if several of them are already under partial or complete testing. For the other categories of components, in general the focus is instead on refinement, testing and integration, while the development is almost finished for all of them. Dealing with Technology Readiness Level (TRL), also given in TABLE IV. , it is possible to appreciate how most of the components already passed the middle of the TRL goal, meaning that they are closer to the TRL expected at the end of the WP (M40) than the one at the beginning of the project (M0). Currently (M30), the minimum achieved TRL is 3, *Proof-of-Concept Demonstrated, Analytically and/or Experimentally*, while the maximum is 6, *System Adequacy Validated in Simulated Environment*.

TABLE IV. STATUS OF THE WP5 COMPONENTS: MAIN ACTIONS IN PROGRESS (DEVELOPMENT, REFINEMENT, TESTING, INTEGRATION) AND TRL IN THE DIFFERENT PROJECT PHASES (@M0, @M30, @M40).

Category	ID	Develop.	Refin.	Testing	Integr.	TRL@M0	TRL@M30	TRL@M40
HW	3.1		x	x	x	2/3	5	5/7
	3.2	x		x	x	4	5	6
	3.3	x				3	3/4	4
	3.4				x	N/A	N/A	N/A
	3.5	x		x		1	3	6
	3.6	x		x		1	3	4
	3.7	x				3/4	5	6
	3.8	x		x		N/A	N/A	N/A
	3.9	x				0/4	5	4/9
	3.10	x		x		1	3	6/7
	3.11	x	x	x		4	5	6
SW	4.1		x	x		1	4	4/6
	4.2		x		x	1	4	4/6
	4.3	x		x		2/3	5	7
	4.4	x		x	x	4	5	6
	4.5			x	x	0	3	5/6
	4.6		x	x	x	4	5	6/7
	4.7	x				3	4	6/7
	4.8		x	x		4/5	5/6	6/7
	4.9	x	x			2	3	5
COMM	5.1	x		x		3	5	5
	5.2		x	x		2	4	6
	5.3					4	6	6/9
	5.4		x		x	4	4	6/7
MISC	6.1	x	x	x	x	0	4	5/6
	6.2				x	2	3	4/5

E. WP6 – Use Cases and Demonstrators

The goal of WP6 is to integrate the developments achieved in the technical WPs. The implementation of these technologies is driven by the different industrial UCs that aim to demonstrate how the reference architecture, and desing- and run-time strategies can be adopted in different industrial scenarios. The whole set of scenarios covers a wide spectrum of industries that ensure the applicability of the FitOptiVis solution to diverse markets such as medicine, traffic control, production lanes or space

applications. The different UCs have been working on the integration of the solutions for the last part of the project, being able to learn lessons and to identify exploitation potentials, considering the specific industries. In the coming paragraphs a focus on each UC will be provided, trying to highlight the mentioned aspects.

1) UC1 – Water Supply

UC1 is focused on distributed and dynamic critical infrastructure control and protection. The final goal of this UC is to develop a monitoring system composed of smart cameras, drones, and heterogeneous sensors that, leveraging on FitOptiVis solutions, can adapt at run-time changing functionalities and optimizing resources and efficiency. UC1 involves video processing algorithms for intrusion detection, face recognition and movement detection; a multi sensors gateway that collects data from sensors, aggregates, processes, and visualizes them, and controls actuators; adaptive HW accelerators with performance optimization capabilities; a drone equipped with onboard camera. All these components are being developed and integrated, and will be installed in a real, available water supply facility. The monitoring application supports functional adaptivity: it has different working points with different trade-offs in terms of video quality, bandwidth, computational resources, and power consumption. A different set of video surveillance functions corresponds to each working point. Leveraging on FitOptiVis approach, at run-time each camera switches among working points, according to the information it or the other cameras has just detected.

Critical infrastructures protection, video surveillance and artificial intelligence processing on the edge are reference markets for UC1. Such markets have reached a huge size and they are constantly growing as critical infrastructure safety, maintenance, reliability, and security are very important, and a priority for every nation. UC1 outcome, that is a demonstrator prototype of an advanced and innovative video surveillance system for a water supply infrastructure, will be exploited in such relevant commercial sectors representing important market opportunities for all the involved partners. Thanks to its features, the developed solution represents a valuable security system for all customers that need to protect a distributed critical infrastructure.

2) UC2 – Virtual Reality

UC2 main goal is to push further virtual reality technology to provide a new multi-camera system capable of reaching a real-time behavior. Within this scope, the UC leader (Nokia) also investigated video data compression solutions. As a result of UC2, it recently presented the world's first implementation and source code release of the upcoming MPEG standard for video-based point cloud compression (V-PCC) on current mobile HW. V-PCC enables offloading most of computational burden to dedicated devices, letting it possible to reach real-time decoding, and it can be distributed and stored easily with existing 2D video technologies (ISOBMFF and DASH). Within UC2, V-PCC compression efficiency has been compared to state of the art and approaches for V-PCC distribution have been studied. Nokia also presented MPEG standard for V-PCC in its latest form, including a decoder implementation for real-time augmented reality playback and performance evaluation.

Many standards to compress images, video, and LIDAR sensor data, are present in literature. The objective of V-PCC is not to compress the raw sensor data, but to compress the point cloud representations of the scenes captured by the sensors. The developed coding techniques are designed to be agnostic of the specific sensors used to create the point cloud data, so it is assumed that, prior to compression, 3D data from different sensors is fused to generate the point cloud representation to be compressed. In terms of markets, there are already some truly revolutionary and challenging ones where the developed technologies can be effective, such as the combination of autonomous cars and augmented/virtual reality, which are more theoretical than practical, or game streaming, which is more realistic and is poised to grow in the near future. Game streaming is a perfect example, combining ultralow latency requirements and high bandwidth with the all-important inclusion of real-time interactivity.

3) UC3 – Habit Tracking

UC3 focuses on the e-Health domain targeting the elderly or dependent people. The objective is to increase their autonomy living alone at home: 1) increasing their safety by monitoring their daily lives to trigger alarms in case of accidents and 2) assessing their current physical habits and identifying the situations where methodological and behavioral concepts can efficiently promote physical activity and healthy lifestyle. FitOptiVis UC3 could help people in their journey towards a healthier and more balanced way of life. The habit tracking system could also help detecting deviations from a standard behavioral pattern.

The core components of the habit tracking system are: 1) local video processing at edge nodes in embedded devices that reach real-time performance; 2) vital sign collection via telemedicine modules; 3) a convergent network that allows different traffics to share the network, ensuring bounded latencies for critical data such as accident alarms in our case.

The requirements for UC3 include state-of-the-art recognition accuracy, low latencies for critical traffics, and smart use of energy for the embedded devices for video processing. FitOptiVis technologies support the development of the UC3 reconfigurable CPS to achieve this multi-objetive optimization continuously monitoring the operational qualities via FIVIS. For example, run-time adaptation at the edge enables: 1) the deployment of complex state-of-the-art Deep Learning video processing offloading part of their complex processing to high-end servers via PoCL-R, when seeking confirmation for an accident to reduce false alarms; 2) the deployment of simpler network architectures with low-energy budgets, when monitoring daily life activities

The global digital health market is expected to keep growing in the next years favored in our case by the current population ageing. The transition of the healthcare industry into digital healthcare systems for management and analysis of patient health is expected to be the most vital driver of the market. The benefits encouraging higher demand include round the clock care service, wider and faster access to patient information, reduction of administrative and medical errors, self-monitoring and management by patients and the centralization of the entire healthcare industry chain.

4) UC4 – 3D Industrial Inspection

FitOptiVis has encouraged the UC4 leaders to improve their industrial inspection system adopting an edge computing architecture. To do so, new requirements had to be defined establishing a set of key performance indicators to check their achievements. Also, a concrete advantage has been taken from the new QRML language developed in the project to model this new architecture. The new system includes algorithms that can be selected depending on the current status of the execution, in other words, the capabilities are adapted during the run-time phase. In addition to this, the monitoring information is sent to FIVIS, another output produced by FitOptiVis which is a monitoring solution to store and visualize data using dashboards. Related to this component, it has been developed an open-source plugin for Telegraf that is being used also by other partners of the consortium. Finally, the new Edge Computing architecture adopted during the project, has allowed the research of new methods for determining the pose of an object using a multicamera system and a patent application was launched.

Many manufacturing industries have developed mature processes to create their products from a well-known set of raw materials, machinery, and tools, but there are still open problems in terms of quality control. The aim of UC4, ITI's industrial inspection system (Zero Gravity 3D), is to demonstrate that with this new solution it can be possible to obtain a considerable increase of productivity. The first main target is spring manufacturers market. The market studies conducted by UC4 leader and market searches in general have shown that manufacturers face similar issues in terms of quality control as well as in pressure on selling prices, and a strong push for shorter delivery time. Manufacturers are focused mainly on fighting these issues working on principles like innovation, technology, and lean operations. The proposed UC4 solution has been designed based on these principles. All the improvements introduced have been oriented to reduce the bandwidth usage and increase the throughput of the inspection device and now, the challenge to solve is to find a mechanism that increases the number of parts that can be thrown through the device. In other words, the computational components of the system are ready for a mechanical engineering evolution increasing the number of parts per minute arriving to the cameras.

5) UC5 – Road Traffic Surveillance

Using FitOptiVis developments in UC5 an object detection engine has been enhanced to make Licence Plate (LP) detection effective. The core of the UC development is low-level FPGA technologies developed within FitOptiVis and LP detection component that has been enhanced using profiling tools. For the run-time, advantage of FIVIS has been taken for effective monitoring of the device running in simulated or real environments. All these development steps led to the effective and bug-free implementation of the smart camera device for traffic surveillance.

Smart camera that has been developed within FitOptiVis project can be exploited in many intelligent traffic surveillance systems including mobile surveillance. These systems can be static, such as section speed control or travel time estimation, or embedded into vehicles, such as car parking monitoring. In the case of static systems, low power consumption and hermetical

sealing with IP rating are of great advantage, as there is often lack of stable power supply and the device needs to be waterproof. In the mobile enforcement, the monitoring vehicle must be equipped with multiple cameras. The processing of the camera streams is very computationally intensive. With the license plate detection component, the computationally intensive part as well as other connected tasks relevant for the application can be done directly in the cameras.

6) UC6 – Multi-Source Streaming Composition

In UC6 the input stage (fiber connection and downscaler) of the compositor is the prototype of a resource limited streaming system. As such it has been used in the original definition of the QRML DSL. Later, specifics distilled from the other UCs were added, and interfaces to solver tools were developed on the top of the original QRML model. This technology is the basis of the run-time resource management in the final medical product. Simulation tools originally developed in the Almarvi project [135] and extended in WP3 of FitOptiVis are used to prove correct functioning of several parts of the compositor and to do buffer size optimizations.

The composition device developed in FitOptiVis will become part of the successor of the FlexVision solution, an option readily available in the Philips Azurion product family [136]. This solution will have substantially lower latency, more flexibility and improved image quality compared to its predecessor. FlexVision plays a central role in the workflow of any clinical procedure, having all relevant information on one screen. About 70% of the systems currently have FlexVision, and this is expected to grow as large screens will become even more important in the near future. The compositor device developed in this UC also provides more flexibility (extra screens) than its predecessor and this new flexibility will contribute to its growth.

7) UC7 – Sustainable Safe MRI

UC7 is focused on modifying an existing MRI system (*Philips Ingenia Elition X* [137]) to let the operator determine the optimal balance between physiologic stresses in patients of several kinds, energy consumption and the resulting quality of the image. Several activities have been developed so far in the frame of the project to achieve a prototype able to consume less energy per scan (WP3 and WP5), predict the behavior of system components with respect to power demand and heat dissipation to patients (WP5 components), and precisely estimate, control, and minimize the thermal load whilst providing superior image quality. A partial demonstrator has been developed, which contains a new gradient coil which can create strong time-varying magnetic field. These fields can create involuntary muscle movement through peripheral nerves stimulation and must be well controlled for safety reasons while in the main time minimizing power. In addition, an RF coil has been designed for emitting RF pulses to the human body. In this case tissue can absorb the RF fields and heat-up when proper software controls are not in place. Design-time models have been created and an abstraction in run-time models for efficient execution, thereby minimizing thermal load, while providing superior image quality has been created and the tested on phantoms. The coming year, design-time models and real-time abstractions will be optimized through scanning on healthy volunteers.

When it comes to exploitation, the goal of the UC7 solution is to put into the market a new MRI system, which will be able to obtain better images than the actual systems in the market. Philips is one of the main actors of the medical technology and the solutions developed within the FitOptiVis project will reach new safer and more sustainable MRI systems.

8) UC8 – Robots Calibration

The robot calibration system considered in UC8 consists of two main components: a point tracker to localize points of interest in 3D world coordinate system, and a mathematical apparatus that estimates robot parameters from a robot model and the point cloud acquired during calibration movements. Then UC8 can be implemented as an image processing and computing pipeline for which control, optimization and management are subject of the FitOptiVis project. WP3 supported accelerators are used in the new version of the object tracker. The whole system is modular, allowing to use prototype components in early stages and replace them with optimized ones later without the need to re-implement the rest of the application. The two components realizing the robot calibration system, part of the FitOptiVis WP5 component library, are currently under refinement. The solution targets middle-level applications where it is not possible to afford high-tech, expensive calibration methods but still precision is of a high concern. It is estimated that manufacturing costs of a system capable to localize points with 0.1 mm accuracy in working area of 1x1x1m will be around 4000 euros.

Commercial activities in scope of UC8 during the project are focused on the robotic calibrator as a whole, and on the point tracking system as its component. As a small company, the UC leader (REX Controls) is aware of the need to look for new

suitable partners, convince them of the quality of the developed solution and meet their requirements. Simultaneously with the identification of potential partners, it is necessary to work on the corresponding product offer and adjust system parameters to the majority needs. REX plans to offer not only the full robot calibration suite, but also the point tracking subsystem, which may have even bigger economic potential than the full robot calibrator. Finally, it is necessary to note that the expected result of this UC is a prototype, with a target TRL of 5/6. It is not expected that the final product will be available during the FitOptiVis project. After extensive testing of the prototype in real environment, its potential will be evaluated and in case of success it will be turned into a real product, defining its way to market.

9) UC9 – Surveillance of Smart-Grid Critical Infrastructure

UC9 is oriented to reliable and remote real-time control and monitoring of a critical infrastructure, such as an electrical substation. The proposed solution can be applied not only to energy critical infrastructures for generation, transmission, and distribution, but also to railway energy control. UC9 consists of a distributed smart video surveillance network fully integrated with ultra-reliable smart-grid critical infrastructure placed at remote electrical substations. The resulting system adapts the substation operation to prevent potential harms on workers or intrusions caused by undesired accesses, while preserving the determinism required by the smart-grid network. All this after the integration in the same network of the video surveillance traffic that demands high data bandwidth. The coexistence of both technologies is sustained by a time sensitive networking based solution delivering latency control and time synchronization to support distributed processing systems and strict real-time control.

To support the hybrid communication, the TSN has implemented IEEE 802.1Q VLAN bridging to provide differentiation and prioritization of the data streams. The hybrid communication along with deterministic QoS is supported by the Time-Aware traffic Shaper of the egressing traffic, fully coordinated by the generalized Precision Time Protocol (gPTP). Also, the smart-grid equipment on the electrical substation is deployed onto a high-availability seamless redundancy network for ultra-reliability. Synchronization and smart-grid traffic are considered critical and thus the highest priority assigned to them, ensuring low bounded latencies and guaranteed bandwidth.

The smart surveillance provides person tracking enforced by biometrical recognition of people in the electrical substation, detecting intruders and tracking authorized personnel to confirm they are carrying assigned tasks correctly. A smart runtime reconfiguration mechanism enables the transmission of high-resolution video triggered by e.g. the detection of an intruder or when personnel enters an unauthorized area. The rest of the time, the multiple cameras of the system are only transmitting low resolution data, greatly reducing the use of bandwidth. Regarding the deterministic network, the detection of intruders trigger alarms that are considered critical. This alarm is used to change the control mode of the electrical substation from local to remote, in order to prevent accidents. The highest priority is required to reduce latencies when transmitting the command through the same network that also is used for the transmission of the video streams from multiple cameras.

UC9 is oriented to reliable and remote real-time control and monitoring of a critical infrastructure, such as an electrical substation. The proposed solution can be applied not only to energy critical infrastructures for generation, transmission, and distribution, but also e.g. to railway energy control, autonomous navigation, or industrial robotic networks.

10) UC10 – Autonomous Exploration

UC10 consists of a video chain capable of identifying satellite models through different possible implementations and set points. So far, in UC10 the following advantages have been obtained from the technologies developed: an extended version of the QRML has been used for building the reference architecture and describing the possible qualities and set points of the HW and SW components that are being used; design- and run- time adaptivity has been improved thanks to the developments on WP3 and WP4 so far. This UC benefits particularly from them when it comes to changing between several configurations that are required on the different stages of the space mission and allow to change between local and remote implementations. Within the frame of FitOptiVis several HW and SW components are being developed that use the specific technologies from the project, such as an image collection interface capable compression component, or an emulator of a ground segment, all of them able of adapting to the mission conditions thanks to the technologies of the project.

As the space industry has very strict requirements due to its limited market capability, the developments of FitOptiVis should go through several stages of validation before they could go into the market. However, the main niche for these developments could be in low-cost applications where these requirements are more relaxed, and where the technologies could be applied as a

way to explore their capabilities in the space sector. In the long term, if they show good results in low-cost applications, they could potentially be used within more expensive missions with tougher requirements.

11) Summary

In this last summary section, we provide in TABLE V. a more direct and complete overview of the FitOptiVis UCs with a clear idea of specific technology issues faced, and the ongoing advancements within the project.

TABLE V. FITOPTIVIS WAY OF ADDRESSING THE UC ISSUES THROUGH THE TECHNOLOGIES UNDER DEVELOPMENT WITHIN THE PROJECT.

UC	**Actual Technology Issues**	**Advancements with FitOptiVis Technologies**
UC1 Water Supply	Classical SCADA systems only provide data gathering/presentation from components with common interfaces.	***Smart integration*** and cooperation of heterogeneous components (e.g., cameras, drones) through ***abstraction*** and ***virtualization***, adopting model-driven design approaches able to guarantee better 1) collected data management, 2) interfacing with a large plethora of complex sensors interfaces.
		Distributed and smart communicating nodes contribute to improve efficiency and reliability, by reducing false alarm rate due to environment unpredictable conditions i.e., weather, temperature, presence of animals, etc.
	Water supply critical infrastructure must avoid water waste, natural disasters, and malicious acts.	Enhancement of system efficiency/reliability through: • improved damage detection algorithms, • drones with sensors/actuators to access unsafe areas, • face recognition techniques to ensure robust access control.
	Classical SCADA systems do not provide adaptability.	***Smart cameras*** for edge processing offering several power, network, and storage trade-offs thanks to ***reconfiguration.***
UC2 Virtual Reality	Depth map algorithms are not equally efficient under different conditions (e.g., indoor/outdoor) or while considering different platform components (sources or computing nodes).	Improvement of the critical depth map computation by advanced video analytics algorithms and system level performance, by mean of the proposed hw-sw **co-design** techniques and **cross-customization** of the algorithms with respect to the available platform components.
		Abstraction of components on a ***virtual reference platform*** allows algorithms customization for resource usage and memory bandwidth minimization while maximizing ***parallel and local processing***.
UC3	Actual systems monitor and	***Smart integration*** and ***components abstraction***

Habit Tracking	generate alarms for specific situations.	leads to system generalization, moving the focus to behavioral patterns, to address life quality improvement rather than surveillance.
	Actual systems cannot manage big amounts of data, real-time constraints, and adaptation.	***Edge processing*** and ***reconfigurable workload distribution,*** built upon ***distributed communicating nodes***, lead to ***real-time intelligent resource management***.
UC4 3D Industrial Inspection	Machine video-based inspection systems are specific, costly, and run computationally intensive algorithms.	***A virtual reference platform*** allowing early optimization and customization leveraging on ***reconfigurable components*** for ***real-time resource managemen*****t** allows system generalization, costs reduction (by means of ***generalization***) and performance improvements (by means of **customization**).
UC5 Road Traffic Surveillance	Current systems do not provide edge processing, resulting in bandwidth and cores overloading.	***Model-driven self-adaptive monitoring approach*** and ***re- configurability support*** to design optimal by design infrastructures capable of ***efficiently (re-)distributing*** the load between ***heterogeneous*** components.
		Deployment of ***advanced embedded power-aware systems*** allowing ***hermetic sealed package*** and ***more reliable higher performance infrastructures***.
UC6 Multi-Source Streaming Composition	Multi-source streaming systems limit composability to isolated images scaling, leading to distortion.	***Transparent integration*** and coordination for ***real-time management*** of image sources and compositor engine, resulting in ***bandwidth reduction, enhanced operability, and efficiency (power, cost, reliability)*** and enabling embedded (FPGA) techniques.
UC7 Sustainable Safe MRI	Devices with digital control allow precise MRI power deposition, but the potential degrees of freedom are only partially exploited.	***Model-driven engineering*** leading to ***composable and predictable*** cross-optimized systems (by design) to precisely estimate, control and minimize the thermal load whilst providing superior image quality.
UC8 Robots Calibration	Robot calibration requires repeatability and non-invasiveness, which today is typically time, money, and resource consuming.	***A model-based approach*** based on ***scalability, interoperability and composability*** enables to ***adapt*** a given processing pipeline to the available robotic systems and the calibration hw-sw resource configuration, resulting in ***lower costs and reduced calibration time***.
UC9 Surveillance of Smart-Grid Critical Infrastructure	Data protection to be ensured, security crew costs to be reduced.	Improvements of video processing algorithms (detection of independently moving objects, active vision models, recognition from motion, etc.)
	Systems isolated and spread	Adoption of ***communication schemes that***

	in remote areas leveraging on conventional constrained network.	***offer timing guarantees*** (TSN) to improve transmission and to integrate mixed criticality data (TSN links sharing), reducing communication costs.
	Efficient and real-time processing of video streams to trigger alarms and reduce false alarms.	***Model-driven engineering*** provides optimal by construction systems implementing ***run-time adaptation across a networked platform*** to achieve several trade-offs among resources, accuracy, and real-time processing.
UC10 Autonomous Exploration	New technologies have great potential, but they are not exploited properly to cope with power, performance, safety, timeliness, security, and reliability requirements of the space scenario.	***Component abstractions*** and ***virtualization*** of ***high- performance data-processing platforms*** help in providing methodologies, with ***accelerator, GPU, and FPGA support***, for the implementation of a ***model-driven monitoring approach that can adapt itself,*** promptly answering to the challenging space constraints.

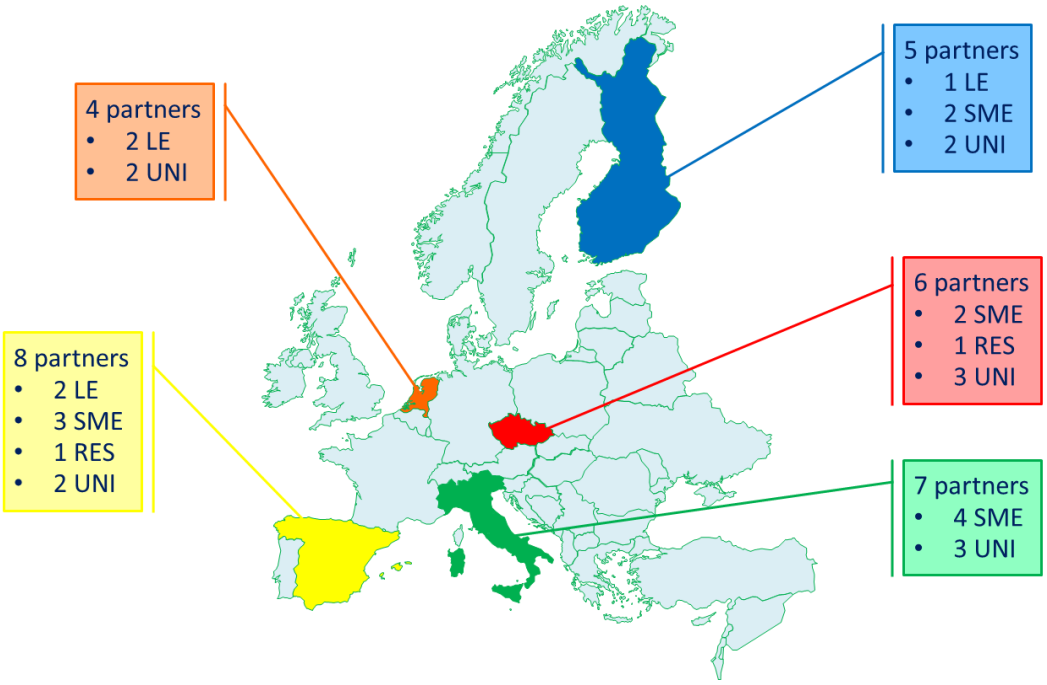

Fig. 11. FitOptiVis Consortium

VIII. Consortium as a Whole

Philips is the project coordinator and has the overall responsibility for the FitOptiVis consortium. This latter is composed of 30 partners from 5 countries (Fig. 11): 5 Large Enterprises (LE), 11 Small and Medium Enterprises (SME), 2 RESearch institutes (RES), 12 UNIversities (UNI). All the FitOptiVis partners are highly active at both European and international levels and exhibit a core set of expertise in contributing towards medium- to large-scale European and national projects.

Several large, small, and medium size enterprises producing CPSs, where image processing is a crucial element of correct execution, are involved in the project. In all cases the CPS has a distributed set of image- and video-sensors whose inputs need

to be processed fast, and in many cases with low energy use. Some partners have other dependability requirements that need to be regarded as well. Other SME partners, RES and UNI provide appropriate innovative technologies and methodologies to tackle QRM for image- and video-processing. The different companies are providing demonstrations involving their own CPS, instantiating the FitOptiVis architecture, run-time support and/or components and applying the design-time technologies. The solid expertise and the cooperation of all the partners together allowed to successfully carry out the project, challenging our ambitious objectives and demonstrating the project results by means of 10 different demonstrators.

IX. Remarks, Plan and Acknowledgements

FitOptiVis is going through its final year, full demonstrators will be available in August 2021 and the third year will lead us to consolidate all the innovative technologies, libraries and tools that have been described in this paper.

This work is part of the FitOptiVis project [1] funded by the ECSEL Joint Undertaking under grant number H2020-ECSEL-2017-2-783162. Several national funding agencies also contributed to the project funding. A special thanks to all the FitOptiVis consortium partners that contributed to the FitOptiVis project on which this paper is based. It is worth noting some ECSEL projects that have provided background and/or reusable results taken into account in FitOptiVis: MegaM@rt2 [138], AQUAS [138], CASPER [139], AFarCloud [141].

References

[1] Al-Ars, Z., et al. (2019). The FitOptiVis ECSEL project: highly efficient distributed embedded image/video processing in cyber-physical systems. ACM International Conference on Computing Frontiers.

[2] Pomante, L., et al. (2020). Design and management of image processing pipelines within CPS: 2 years of experience from the FitOptiVis ECSEL Project. Euromicro Conference on Digital System Design.

[3] Cyber-Physical Systems. Online (accessed: 28/01/2021): http://cyberphysicalsystems.org/

[4] van den Berg, F., et al. (2020). QRML: A Component Language and Toolset for Quality and Resource Management. Forum on specification & Design Languages.

[5] Quality and Resource Management Language. Online (accessed: 28/01/2021): https://qrml.org

[6] Hendriks, M., et al. (2020). Interface Modeling for Quality and Resource Management, Logical Methods in Computer Science.

[7] Alizadeh Ara, H., et al. (2018). Compositional Dataflow Modelling for Cyclo-Static Applications. Conferenece on Digital System Design.

[8] Moro, A., et al. (2018). Hardware performance sniffers for embedded systems profiling. International Workshop on Intelligent Solutions in Embedded Systems.

[9] AUTOSAR. Online (accessed: 28/01/2021): https://www.autosar.org/

[10] ARINC Specification 651. Design Guidance for Integrated Modular Avionics, (1991).

[11] ARINC specification 653. Avionics application software standard interface, (2010).

[12] Tripakis, S. (2016). Compositionality in the Science of System Design. Proc. IEEE.

[13] Masrur, A., et al. (2016) Component-Based Design of Cyber-Physical Applications with Safety- Critical Requirements. Microprocessors and Microsystems.

[14] Bures, T., et al. (2016). Software Abstractions for Component Interaction in the Internet of Things, Computer.

[15] Guo, J., et al. (2013). Variability-Aware Performance Prediction: A Statistical Learning Approach. IEEE/ACM International Conference on Automated Software Engineering.

[16] Hu, Y.C., et al. (2015). ETSI White Paper No. 11, Mobile Edge Computing, A key technology towards 5G.

[17] 5G Architecture White Paper. Online (accessed: 28/01/2021): https://5g-ppp.eu/wp-content/uploads/2018/01/5G-PPP-5G-Architecture-White-Paper-Jan-2018-v2.0.pdf

[18] Yang, B., et al. (2016). Seamless Support of Low Latency Mobile Applications with NFV-Enabled Mobile Edge-Cloud. IEEE International Conference on Cloud Networking.

[19] Geilen, M., et al. (2007). An Algebra of Pareto Points. Fundamenta Informaticae.

[20] Shojaei, H., et al. (2013) A Fast and Scalable Multi-dimensional Multiple-choice Knapsack Heuristic. ACM Transactions on Design Automation of Electronic Systems.

[21] Lee, E. (2015). The Past, Present and Future of Cyber-Physical Systems: A Focus on Models. Sensors.

[22] Balasubramanian, D., et al. (2014). Taming Multi-Paradigm Integration in a Software Architecture Description Language. Workshop on Multi-Paradigm Modeling.

[23] Hutchinson, J., et al. (2014). Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure. Science of Computer Programming.

[24] Jääskeläinen, P., et al. (2016). POCL: A Performance-Portable OpenCL Implementation. International Journal on Parallel Programming.

[25] Mittal, S. (2015). A Survey Of Techniques for Architecting and Managing Asymmetric Multicore Processors. ACM Computing Surveys.

[26] Mittal, S. et al. (2015). A Survey of CPU-GPU Heterogeneous Computing Techniques. ACM Computing Surveys.

[27] Dyplo. Online (accessed: 28/01/2021): https://topicembeddedproducts.com/products/dyplo/

[28] Xilinx Vivado Design Suite. Online (accessed: 28/01/2021): https://www.xilinx.com/products/design-tools/vivado.html

[29] Palumbo, F., et al. (2017). Power-Awarness in Coarse-Grained Reconfigurable Multi-Functional Architectures: a Dataflow Based Strategy. Journal of Signal Process. Systems.

[30] Geilen, M., et al. (2017). Performance Analysis of Weakly-Consistent Scenario-Aware Dataflow Graphs. Signal Processing Systems.

[31] Sau, C., et al. (2014). Automated design flow for coarse-grained reconfigurable platforms: An RVC-CAL multi-standard decoder use-case. International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation.

[32] Grüttner, K., et al. (2017). CONTREX: Design of embedded mixed-criticality CONTRol systems under consideration of Non-functional properties, Microprocessors and Microsystems.

[33] Reconfigurable Platform Composer Tool (RPCT) Project. Online (accessed: 28/01/2021): https://sites.unica.it/rpct/

[34] Pomante, L., et al. (2020). SystemC-based electronic system-level design space exploration environment for dedicated heterogeneous multi-processor systems, Microprocessors and Microsystems.

[35] Anwar A., et al. (2014). On the use of microservers in supporting hadoop applications. International Conferenece on Cluster Computing.

[36] Jung, Y., et al. (2012). A broadband embedded computing system for MapReduce utilizing Hadoop. International Conference on Cloud Computing Technology and Science.

[37] Gibert, E., et al. (2015). Profiling Support for Runtime Managed Code: Next Generation Performance Monitoring Units. Computer Architecture Letters.

[38] Tuveri, G., et al. (2016). On-the-fly Adaptability for process networks over shared-memory platforms. Microprocessors and Microsystems.

[39] Sinha, S., et al. (2015). Composable and predictable dynamic loading for time-critical partitioned systems on multiprocessor architectures. Microprocessors and Microsystems.

[40] Multanen, J., et al. (2016). OpenCL Programmable Exposed Datapath High Performance Low-Power Image Signal Processor. Nordic Circuits and Systems Conference.

[41] Renganarayana, L., et al. (2012). Programming with relaxed synchronization. ACM workshop on Relaxing synchronization for multicore and manycore scalability.

[42] Zemcik, P. et al. (2011). AX32 Low Power Embedded Video Enabled System Using FPGA. Conference on Field Programmable Logic and Applications.

[43] Imran, N., et al. (2015). Distributed video coding for wireless video sensor networks: a review of the state-of-the-art architectures. SpringerPlus.

[44] Janabi-Sharifi, F. , et al. (2010). A Kalman-Filter-Based Method for Pose Estimation in Visual Servoing, IEEE Transactions on Robotics.

[45] Marshall, M., et al. (2014). Kalman Filter Visual Servoing Control Law. IEEE International Conference on Mechatronics and Automation.

[46] Pauwels, K., et al. (2013). Real-time model-based rigid object pose estimation and tracking combining dense and sparse visual cues. IEEE Conference on Computer Vision and Pattern Recognition.

[47] Hu, F., et al. (2017). Cyberphysical System With Virtual Reality for Intelligent Motion Recognition and Training. IEEE Transactions on Systems, Man, and Cybernetics: Systems.

[48] Lukman Khalid, C.M., et al. (2014). Integration of cyber-physical systems technology with augmented reality in the pre-construction stage. Conference on Technology, Informatics, Management, Engineering & Environment.

[49] Rhee, T., et al. (2017). MR360: Mixed Reality Rendering for 360°; Panoramic Videos. IEEE Transactions on Visualization and Computer Graphics.

[50] Philips FlexVision. Online (accessed: 28/01/2021): https://www.usa.philips.com/healthcare/product/HCOPT15/flexvisionxldisplayscreen

[51] Axis introduces video analytics with reliable, easy toinstall AXIS Guard Suite for enhanced videosurveillance efficiency. Online (accessed: 28/01/2021): https://www.axis.com/files/press_releases/se/pr_4025_en_se_20150928.pdf

[52] Bosch video analytics at the edge. Online (accessed: 28/01/2021): https://www.boschsecurity.com/xc/en/solutions/video-systems/video-analytics/video-analytics-at-the-edge/

[53] Garibotto, G.B., et al. (2015). Object Detection and Tracking from Fixed and Mobile Platforms. Image Analysis and Processing.

[54] Barranco, F., et al. (2012). Parallel architecture for hierarchical optical flow estimation based on FPGA. IEEE VLSI Systems.

[55] Rodriguez, R., et al. (2015). Codebook hardware implementation on FPGA for background subtraction. Journal of Real-Time Image Processing.

[56] Sardar, S., et al. (2014). Hardware Implementation of Real-Time, High Performance, RCE-NN based Face Recognition System. VLSI Design Conference.

[57] Meloni, P., et al. (2016). A high-efficiency runtime reconfigurable IP for CNN acceleration on a mid-range all-programmable SoC. International Conference on ReConFigurable Computing and FPGAs.

[58] Yviquel, H., et al. (2015). Embedded Multi-Core Systems Dedicated to Dynamic Dataflow Programs. Journal of Signal Processing Systems.

[59] Hoozemans, J., et al. (2017). VLIW-based FPGA computational fabric with streaming memory hierarchy for medical imaging applications. International Symposium on Applied Reconfigurable Computing.

[60] Janke, R., et al. (2014). Protecting Water Supply Critical Infrastructure: An Overview. Chapter 2, Securing Water and Waste Water Systems.

[61] Doro-on, A. (2011). Risk Assessment for Water Infrastructure Safety and Security. CRC Press.

[62] Kroll, D.J. (2006). Securing Our Water Supply: Protecting a Vulnerable Resource. PennWell.

[63] EPANET – Application for Modeling Drinking Water Distribution Systems. Online (accessed: 28/01/2021): https://www.epa.gov/water-research/epanet

[64] Guo, H., et al. (2015). A novel approach for global abnormal event detection in multi-camera surveillance system. IEEE International Conference on Information and Automation.

[65] Fahmy, M. (2009). Detecting and locating leaks in Underground Water Mains Using Thermography. International Symposium on Automation and Robotics in Construction.

[66] Shakmak, B. (2015). Detection of Water Leakage in Buried Pipes Using Infrared Technology. IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies.

[67] Cheon, L., et al. (2011). 3d scene capturing using stereoscopic cameras and a time-of-flight camera. IEEE Transactions on Consumer Electronics.

[68] Velodyne. Hdl-g4e. Online (accessed: 28/01/2021): https://velodynelidar.com/products/hdl-64e/

[69] Skolnik, M.I., (1962). Introduction to radar. Radar Handbook.

[70] Kinect for Windows. Online (accessed: 28/01/2021): https://developer.microsoft.com/windows/kinect/

[71] Occipital Structure sensor. Online (accessed: 28/01/2021): https://structure.io/

[72] Scharstein, D., et al. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International journal of computer vision.

[73] Arcelus, A., et al. (2007). Integration of smart home technologies in a health monitoring system for the elderly, technology from the perspective of the potential occupants. International Conference on Advanced Information Networking and Applications.

[74] Korhonen, I., et al. (2003). Health monitoring in the home of the future, IEEE Engineering in medicine and biology magazine.

[75] Scheutz, M., et al. (2004). Fast, reliable, adaptive, bimodal people tracking for indoor environments, IEEE/RSJ International Conference on Intelligent Robots and Systems.

[76] Fleuret, F., et al. (2008). Multicamera people tracking with a probabilistic occupancy map. IEEE transactions on pattern analysis and machine intelligence.

[77] Tomasi, M., et al. (2010). A novel architecture for a massively parallel low level vision processing engine on chip. IEEE Conference on Industrial Electronics.

[78] NETATMO indoor security. Online (accessed: 28/01/2021): https://www.netatmo.com/security/cam-indoor

[79] Just Checking. Online (accessed: 28/01/2021): https://justchecking.co.uk/peace-of-mind-for-friends-and-family-of-dementia-sufferers-launch-of-just-checking-for-consumers

[80] Graciá, L., et al. (2010). Iterative calibration of multi-view 3D environments using discrete simple targets. Workshop de Reconocimiento de Formas Y Análisis de Imágenes.

[81] Salvador, I., et al. (2011). Computer vision applications: Home, Office and Industry. International Workshop on Interactive Multimodal Pattern Recognition in Embedded Systems.

[82] Cano, J., et al. (2001). Textile inspection with a parallel computer cluster. International Conference on Quality Control By Artificial Vision.

[83] MicroStudio. Online (accessed: 28/01/2021): https://www.microstudio.eu/en_US/page/homepage

[84] FARO. Online (accessed: 28/01/2021): https://www.faro.com/

[85] Salvador, I., et al. (2011). I3D, a system for real-time multi-view 3d inspection. European Automotive Congress.

[86] Salvador, I. et al. (2011). Quality evaluation in a 3D inspection system International Workshop on Interactive Multimodal Pattern Recognition in Embedded Systems.

[87] ARTIS large Display – changing the view of interventions. Online (accessed: 28/01/2021): https://www.siemens-healthineers.com/it/angio/options-upgrades/large-display

[88] BARCO Nexxis. Online (accessed: 28/01/2021): https://www.barco.com/en/product/nexxis

[89] van Helvoort, M. (2014). Feedback-based digital RF Power System. Smart Systems Integration.

[90] van Helvoort, M. (0142). Faster and more efficient thanks to model-driven design. Link Magazine.

[91] EMC^2 Project. Online (accessed: 28/01/2021): https://www.artemis-emc2.eu/

[92] Bastijn,V. (2015). Mixed Criticality Medical Imaging – co-location of real-time and non-real time processing on a single chip/board- HiPEAC.

[93] Hexagon Leica Absolute Tracker. Online (accessed: 28/01/2021): https://www.hexagonmi.com/en-GB/products/laser-tracker-systems/leica-absolute-tracker-at960

[94] DENSO Robotics. Online (accessed: 28/01/2021): https://www.densorobotics.com/

[95] Hayes, B. (2012). The surveillance-industrial complex. Routledge handbook of surveillance studies.

[96] Salvador, I., et al. (2010). Demonstrations of computer vision applications. International Workshop on Interactive Multimodal Pattern Recognition in Embedded Systems.

[97] Sauter, J., et al. (2009). Demonstration of swarming control of unmanned ground and air systems in surveillance and infrastructure protection. IEEE Technologies for Homeland Security.

[98] Kehrer, S. et al. (2014). A comparison of fault-tolerance concepts for IEEE 802.1 Time Sensitive Networks (TSN). Emerging Technology and Factory Automation.

[99] Barranco, F., et al. (2014). Real-time visual saliency architecture for FPGA with top-down attention modulation. IEEE Transactions on Industrial Informatics.

[100] AirRobotics. Online (accessed: 28/01/2021): https://www.airoboticsdrones.com/

[101] Airborne Drones. Online (accessed: 28/01/2021): https://www.airbornedrones.co/

[102] Bajracharya, M., et al. (2008). Autonomy for Mars Rovers: Past, Present, and Future. Computer.

[103] Schuster, M.J., et al. (2016). The LRU Rover for Autonomous Planetary Exploration and Its Success in the SpaceBotCamp Challenge. International Conference on Autonomous Robot Systems and Competitions.

[104] Muttillo, V., et al. (2018). Hepsycode-RT: A real-time extension for an ESL HW/SW co-design methodology. ACM International Conference Proceeding Series.

[105] IVIS-CORE. Online (accessed: 28/01/2021): https://github.com/smartarch/ivis-core

[106] de Moura, L., et al. (2008). Z3: An Efficient SMT Solver. International Conference on Tools and Algorithms for the Construction and Analysis of Systems.

[107] QRMLvis. Online (accessed: 28/01/2021): https://github.com/smartarch/qrmlvis

[108] Goossens, K.G.W., et al. (2017). NoC-based multiprocessor architecture for mixed-time-criticality applications. Handbook of hardware/software codesign.

[109] Tabatabaei Nikkhah, S., et al. (2020). A Performance Analysis Framework for Real-Time Systems Sharing Multiple Resources. Design, Automation, and Test in Europe Conference.

[110] Single source design framework. Online (accessed: 28/01/2021): http://umlmarte.teisa.unican.es/

[111] The Hepsycode project. Online (accessed: 28/01/2021): http://www.hepsycode.com/

[112] Sage repository. Online (accessed: 28/01/2021): https://gitlab.sagelab.it/sage

[113] Narizzano, M., et al. (2019). Property specification patterns at work: verification and inconsistency explanation. Innovations in Systems and Software Engineering.

[114] IMACS. Online (accessed: 28/01/2021): http://www.es.ele.tue.nl/~dgoswami/imacs

[115] TTA-based Co-design Environment (TCE) tools. Online (accessed: 28/01/2021): http://openasip.org

[116] Runtime reconfiguration Implementation of Embedded systems (RIE). Online (accessed: 28/01/2021): https://s3d.unican.es/rie/

[117] Multi-Dataflow Composer design suite. Online (accessed: 28/01/2021): https://github.com/mdc-suite/mdc

[118] Palumbo, F., et al. (2017). Power-Awarness in Coarse-Grained Reconfigurable Multi-Functional Architectures: a Dataflow Based Strategy. Journal of Signal Processing Systems.

[119] Sau, C., et al. (2015). Reconfigurable coprocessors synthesis in the MPEG-RVC domain. Conference on ReConFigurable Computing and FPGAs.

[120] Carta, N., et al. (2013). A coarse-grained reconfigurable wavelet denoiser exploiting the multi-dataflow composer tool. Conference on Design and Architectures for Signal and Image Processing.

[121] Meloni, P., et al. (2018). NEURAghe: Exploiting CPU-FPGA Synergies for Efficient and Flexible CNN Inference Acceleration on Zynq SoCs. ACM Transactions on Reconfigurable Technology Systems.

[122] Kadlec, J., et al. (2019). App note and Eval package. Online (accessed: 28/01/2021): http://sp.utia.cz/index.php?ids=results&id=FitOptiVis-te0726-SDSoC-2018_2

[123] Kadlec, J., et al. (2019). App note and Eval package. Online (accessed: 28/01/2021): http://sp.utia.cz/index.php?ids=results&id=FitOptiVis-te0808-SDSoC-2018_2

[124] Bulej, L., et al. (2021). Managing latency in edge–cloud environment. Journal of Systems and Software.

[125] D'Andrea, G., et al. (2019). Self-adaptive loop for CPSs: is the Dynamic Partial Reconfiguration profitable? Mediterranean Conference on Embedded Computing.

[126] Valente, G., et al. (2020). Dynamic Partial Reconfiguration Profitability for Real-Time Systems. IEEE Embedded Systems Letters.

[127] Palumbo, F., et al. (2019). Hardware/software self-adaptation in CPS: the CERBERO project approach, International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation.

[128] Valente G., et al. (2016). A Flexible Profiling Sub-System for Reconfigurable Logic Architectures. Conference. on Parallel, Distributed, and Network-Based Processing.

[129] Valente, G. et al. (2020). Layering the monitoring action for improved flexibility and overhead control: work-in-progress. International Conference on Hardware/Software Codesign and System Synthesis.

[130] D'Andrea, G., et al. (2020). Work-In-Progress: Cyber-Physical Systems and Dynamic Partial Reconfiguration Scalability: opportunities and challenges. IEEE Real-Time Systems Symposium.

[131] Portable Computing Language. Online (accessed: 28/01/2021): http://portablecl.org

[132] OpenMP. Online (accessed: 28/01/2021): http://www.openmp.org

[133] Deniz, D., et al. (2020). Reconfigurable Cyber-Physical System for Lifestyle Video-Monitoring via Deep Learning. IEEE International Conference on Emerging Technologies and Factory Automation.

[134] FitOptiVis Deliverable 5.1. Online (accessed: 28/01/2021): https://fitoptivis.eu/wp-content/uploads/2019/09/FitOptiVis-D5.1.pdf

[135] Almarvi Project. Online (accessed: 28/01/2021): http://almarvi.eu/

[136] Philips Azurion. Online (accessed: 28/01/2021): https://www.usa.philips.com/healthcare/resources/landing/azurion#page=1

[137] Philips Ingenia Elition X. Online (accessed: 28/01/2021): https://www.usa.philips.com/healthcare/product/HC781358/ingenia-elition-30t-x

[138] Afzal, W., et al. (2018). The MegaM@Rt2 ECSEL project: MegaModelling at Runtime – Scalable model-based framework for continuous development and runtime validation of complex systems. Microprocessors and Microsystems.

[139] Pomante, L., et al (2019). The AQUAS ECSEL Project - Aggregated Quality Assurance for Systems: Co-Engineering Inside and Across the Product Life Cycle. Microprocessors and Microsystems.

[140] Liotou, E., et al. (2020). The CASPER user-centric approach for advanced service provisioning in mobile networks, Microprocessors and Microsystems.

[141] Castillejo, P., et al. (2020). Aggregate Farming in the Cloud: the AFarCloud ECSEL Project. Microprocessors and Microsystems.