Immersive Video Services at the Edge: an Energy-Aware Approach

Pietro Paglierani
Italtel
Castelletto, Milan, Italy
e-mail: pietro.paglierani@italtel.com

Claudio Meani
Italtel
Castelletto, Milan, Italy
e-mail: claudio.meani@italtel.com

Antonino Albanese
Italtel
Castelletto, Milan, Italy
e-mail: antonino.albanese@italtel.com

Paolo Secondo Crosta
Italtel
Castelletto, Milan, Italy
e-mail: paolosecondo.crosta@italtel.com

***Abstract*—To respond to the users' demand for immersive and personalized media services, the 5G and the Multi-access Edge Computing initiatives are proposing novel network architectures. In this context, we present the Video Transcoding Unit (VTU) system, which exploiting the Cloud Enable Radio Access Network proposed by the EU 5G-PPP Sesame project, brings immersive video functionalities to the edge of networks, thus greatly improving User Experience with mobile terminals. A use case is discussed, in which the VTU is deployed in a Stadium or in a large public venue during a Crowded Event, to offer Immersive Video Services. In the proposed architecture, the VTU video processing component can be implemented as a Software-only Virtual Network Function running on different Hardware platforms (X86 or ARM architectures), eventually accelerated by a Graphics Processing Unit. Specific tests are described and discussed and specific Key Performance Indicators are introduced, showing the benefits of the Hardware-accelerated implementation, both in terms of computing performance and of energy efficiency. We believe that the proposed VTU framework significantly advances the state of the art in the provision of video services to the mobile users.**

Keywords-NFV; MEC; 5G; HW acceleration; GPU; Video transcoding.

I. INTRODUCTION

This paper presents an extended and improved version of [1], where we introduced a framework for enhanced video services at the network edge.

The recent worldwide explosion of mobile data traffic in telecommunication networks has been impressive, and this trend will certainly continue in the coming years [2]. The fast spreading of smart terminals and new services based on High-Definition (HD) video have been the main trigger of this explosion, revealing various weaknesses in the architectural and technological approach adopted so far in the design of the traditional mobile network infrastructure [3][4].

The telecommunication market, previously dominated by voice traffic and text messages, is rapidly shifting to completely different and far more complicated scenarios, with millions of connected applications, where even new actors have made their appearance, like machines and "things" (smart home gadgets, vehicles, drones, robots, also including sensors and actuators).

Internet and communication networks have become crucial for any evolutionary process of modern societies and economies. This fact has led to the definition of a new kind of infrastructure based on the "fifth generation" - 5G - architecture, as a response to the requirements coming from the more diverse fields of the future world [3].

5G aims at assuming a fundamental role in the new society; it is not only a simple evolution of previous mobile networks – as was the passage from 3G to 4G - but it stands as a real revolution, able to create the appropriate ecosystem for technical and business innovation [4].

From the technological point of view, 5G will take advantage of the experience coming from the recent convergence of the telecom world with Information Technology (IT). This shift has addressed the necessity coming from Network Operators of lowering general costs, achieving better scalability and reducing the deployment time of new services, and has resulted in a new architectural vision based on Software-Defined Networking (SDN) and Network Function Virtualization (NFV) [5]. 5G will bring the SDN and NFV concepts into the radio communication environment and will use them in a new architectural framework where Multi-access Edge Computing (MEC) will play a major role.

MEC Technology and Architecture concepts are a way to improve both Efficiency and User Experience for a certain number of services. MEC is an ETSI initiative that leverages SDN and NFV principles to push network functions, services and contents to the edge of the mobile network [6].

MEC servers should be directly attached to the base station, but this is not a strict rule because, in this regard, the MEC guidelines are widely open. They provide local computing, storage and networking resources that are virtualized and shared by multiple virtual machines.

Traditionally, all data traffic originating in data centers is forwarded to the mobile core network. The traffic is then routed to a base station that delivers the content to the mobile

devices. In the mobile edge scenario, MEC servers take over some or even all of the tasks originally performed in a data center. Being located at the edge, they eliminate the need of routing data through the core network, lowering communication latency. As such, the MEC paradigm can help to reach the severe requirements posed by 5G in terms of throughput, latency, scalability and automation. However, it's important to note that many of the concepts that are at the basis of MEC and the advantages they bring to a broad range of services are valid regardless of 5G technology (in fact MEC concepts can be similarly applied to fixed networks) and can be demonstrated prior to the coming 5G, for instance by using the well-known WIFI access technology. Though, in view of a wireless access solution fully integrated with the core mobile network, many efforts are now being devoted to combine the MEC principles with the 5G architecture.

In this context, the H2020 5GPPP SESAME project [7] has developed a proposal for Cloud-Enabled Radio Access Network (CE-RAN) systems, based on the evolution of the 4G Small Cell (SC), towards the so-called Cloud Enabled Small Cell (CESC). Such a novel architecture, besides improving radio-related capabilities, can also enable the use of Network Functions Virtualization (NFV) and Software Defined Networking (SDN) at the network edge [5][6]. This goal is achieved through the introduction of the so-called Light Data Center, i.e., an aggregated pool of local and virtualized IT resources, including various types of HW accelerating devices, available to a cluster of CESCs. As a consequence, applications can be deployed at the network edge, implemented as Virtual Network Functions running in the Light Data Center, and thus exploiting the HW acceleration capabilities that the Light Data Center can make available.

Many services can benefit from being hosted at the network edge. Several use cases have been defined in the specification of MEC architecture to demonstrate the advantages of the introduced concepts [6]. One of these use cases regards video services in stadiums and/or large public venues, where video signals created during a sport event or a concert are routed to a MEC server responsible for their local distribution, without involving backhaul connection to the core network. The video contents are also stored in this edge platform, and can be locally processed by applications running on the same MEC server, to create new services and improve User Experience.

The possibility to create, share or receive low-latency, high definition video contents, anywhere and with any device, and with real-time interaction with the system, is usually referred to as Immersive Video Service (IVS). Immersive video applications are attracting a lot of interest, but they still remain very critical functionalities, due to the huge needs of compute, storage and networking resources that HD video brings about.

This paper presents the activities carried out by Italtel Research Labs within the H2020 Sesame project, which led to the development of the VTU system for IVSs. Leveraging MEC principles, the VTU Virtual Network Function (VNF) runs in the Light Data Center (as envisioned by the Sesame architecture), and can thus bring several innovative functionalities to the network edge, greatly improving User Experience with mobile terminals. In particular, the VTU can speed up sharing of Video contents, reduce latency and contribute to increasing the battery life of connected devices offloading them from heavy transcoding operations.

This paper shows how the VTU VNF running in the Sesame CESC Light Data Center can fit in a real use case foreseen by 5G and MEC, and enhance it with some novel features. Also, the limitations of a SW-only implementation with respect to a GPU-accelerated one are highlighted and discussed. The paper provides Key Performance Indicators (KPIs), which can effectively summarize the performance of the VTU, both in terms of compute capabilities, and in terms of energy consumption. Such KPIs can be used to select the most appropriate platform for the specific VTU application context.

In the analysis, Intel X86 architectures with and without GPU acceleration, and ARM architectures are considered, and are used to run tests specifically designed to thoroughly characterize the VTU overall performance.

The paper is organized as follows. Section II presents some related work. Section III briefly describes the activities and the outcome of the Sesame project, in particular in terms of its achievements related to the 5G architecture and the CESC concept. Section IV provides a functional description of the overall VTU system. Section V shows a possible use case for the VTU during localized crowded events. Section VI discusses HW acceleration for video functions, and describes different possible approaches, with their pros and cons. Section VII presents the experimental compute performance characterization of VTU, running on different architectures (x86 and ARM), with or without GPU. Finally, Section VIII summarizes the main results of this work.

II. Related Work

The framework proposed in this paper is based on the architectural results achieved in the Sesame project, in particular on the CESC concept. A general overview of the Sesame approach and of the CESC architecture is summarized is in [8][9]. A solution for the placement of processing and storage capabilities close to the users and a discussion of the advantages of hardware accelerators within the Light Data Center is discussed in [10]. Leveraging such concepts, this paper gives a detailed description of a novel software framework to provide enhanced video services at the network edge.

In the proposed framework, the video transcoding capability plays a key role. The subject of real time video transcoding, in contrast to a batch-oriented approach, is addressed in [11] with the proposition of a video transcoding architecture based on a heterogeneous environment. In [12], a NFV-based MEC platform for a traditional distribution service is proposed, showing the advantages of edge computing platforms in term of bandwidth and Quality of Experience, compared to centralized infrastructures located at the network core.

The topic of power consumption for NFV-based multimedia content delivery is faced in [13], which demonstrates that energy efficiency aspects are as important

as flexibility and performance in the development of VNFs. However, the video transcoding problem and the use of GPUs to accelerate the most compute-intensive video processing workloads is not addressed.

Previous works carried out by the authors, such as [14][15], were related to GPU utilization in NFV environments, and to hardware and software acceleration at the edge of the network, while [16] and [17] discuss the use of GPUs to accelerate a specific video encoding scheme, namely the google VP8 Video encoder.

The present paper goes beyond such results; in particular, it presents a novel and complete solution for IVS in challenging environments such as crowded events, and provides an energy efficiency analysis of the video transcoding process, comparing the performance of ARM-based and X86-based CPUs. Moreover, it analyzes and highlights the benefits of adding GPU resources, to accelerate video transcoding.

III. THE SESAME CLOUD ENABLED SMALL CELL

The 5G-PPP is a joint initiative between the European Commission and the European ICT industry, to drive the activities in the development of the 5G ecosystem. The objective is designing and implementing solutions, architectures, technologies and standards for the next generation communication infrastructure.

The European Union has funded 19 research projects under the 5G-PPP Phase 1 program. Among these, the SESAME Project (Grant Agreement No.671596) introduces three innovative elements in 5G:

- The "placement" of network applications at the network edge, leveraging Network Functions Virtualization (NFV) and Edge Cloud Computing.
- The evolution of the SC architecture, which will play a fundamental role in the 5G infrastructure.
- The consolidation of multi-tenancy in communication infrastructures, thus allowing different operators/service providers to share access capacity and edge computing resources.

In particular, the SESAME project aims at evolving the SC architecture, towards the CESC. This way, virtualized compute capabilities and applications are brought to the network edge, based on the NFV paradigm.

A CESC is an enhanced SC that integrates a virtualized execution platform (micro server) equipped with IT resources (RAM, CPU, storage). A simplified model of the CESC architecture is shown in Fig. 1. With these capabilities, a CESC can support novel applications and services at the network edge.

The basic role in the evolution of the SC concept is played by a specific VNF, namely the so-called SC VNF. The SC VNF represents the link between the radio and the cloud domains. In fact, it can intercept and perform encapsulation/de-capsulation of the S1 interface user data between the Long Term Evolution (LTE) base station component (the so-called eNodeB) and the Evolved Packet Core [7][18]. This way, the user data can be processed by standard VNF, running at the edge in the Light Data Center.

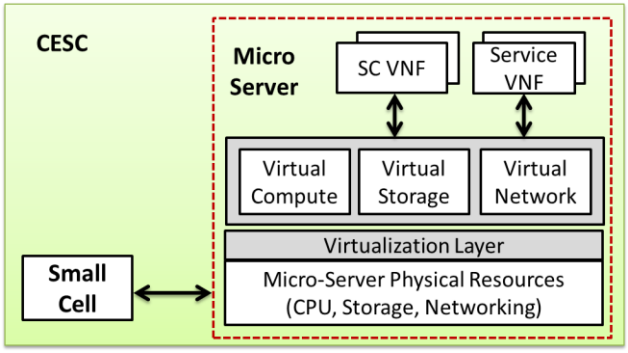

Figure 1. Simplified model of the CESC.

In its basic deployment, the CESC must be able to run the SC VNF, and eventually other VNF, which can run on the low power, low cost IT resources made available to the CESC in its minimal configuration. However, many innovative services, such as IVSs, usually involve compute intensive workloads, which can also require the use of specialized HW accelerators [14][15].

When the basic CESC resources are insufficient to offer the required services, they must be properly enhanced. SESAME proposes the creation of a distributed data center, denominated Light Data Center (Light DC), aimed to enhance the virtualization capabilities and processing power at the network edge.

The Light DC can include HW acceleration devices, such as Graphics Processing Units (GPU), Digital Signal Processors (DSP), and/or Field-Programmable Gate Arrays (FPGA).

Fig. 2 shows a simplified model of the Light DC, where heavy workloads can be offloaded to the additional compute resources. Such resources can be used by VNFs to carry out even compute intensive workloads. However, to offer effective solutions at competitive costs, it is necessary to thoroughly analyze the performance of the proposed VNFs, when executed on the different platforms that can be available at the Light DC.

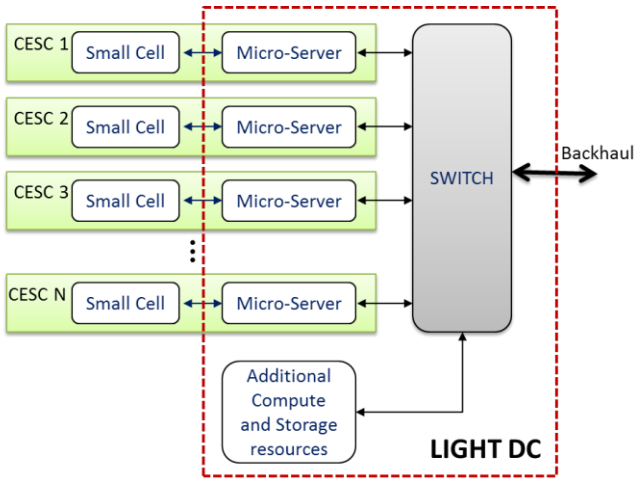

Figure 2. Simplified view of the Light DC Architecture

IV. VTU DESCRIPTION

The VTU framework, developed for IVSs at the network edge, consists of two main components, the VTU VNF and the VTU App. The VTU VNF mainly provides optimized video processing functionalities, which are the main building block to create more complex video services. Furthermore, it can also manage users and groups, by storing and maintaining updated the related information in the so-called VTU User and Group Database.

The VTU App can be downloaded by the users onto their devices; it offers the possibility to register to the VTU system, and create groups of users who can share video contents, exploiting the functionalities made available by the VTU VNF. The main features of the VTU VNF and the VTU App, as well as their interactions, are briefly summarized in the following paragraphs.

A. The VTU VNF

The VTU VNF consists of two VNF Components (VNFCs), namely the Media Engine (ME) and the User and Group Database Manager (UGDM). The ME can provide three basic functionalities, which can be summarized as follows:

- Video transcoding capabilities;
- Video streaming capabilities;
- Local storage for video file upload/download.

Besides video, the VTU ME can also perform audio processing, to provide multimedia services to the users. Though, audio processing functions are not as compute intensive as video functions; hence, for the sake of brevity, this paper will focus only on video capabilities. Finally, the VTU ME can also offer system monitoring functionalities, to support the system maintenance process.

The UGDM VNFC, conversely, can handle and store all the needed information about participants of the CE, which have registered to the VTU system.

1) The VTU ME

The VTU ME can convert video streams from one video format to another. Depending on the type of application that should be provided, the source video stream could originate from a file within a storage facility, as well as coming in form of packetized network stream. Moreover, the requested transcoding service could be mono-directional, as in video stream distribution-like applications, or bi-directional, like in videoconferencing (see Fig. 3).

In the VTU ME, the audio and video transcoding capabilities are provided by the Libav library [19], a very popular open source library, which can perform audio/video processing according to a wide set of coding standards. The AVConv tool from Libav is used for performing the conversion between audio and video formats and containers; while it already supports a wide variety of hardware accelerations, native GPU support in encoding tasks is quite limited, experimental and restricted only to H.264 and H.265 standards, exploiting the NVidia NVENC hardware encoder of medium and high level NVidia GPUs [20].

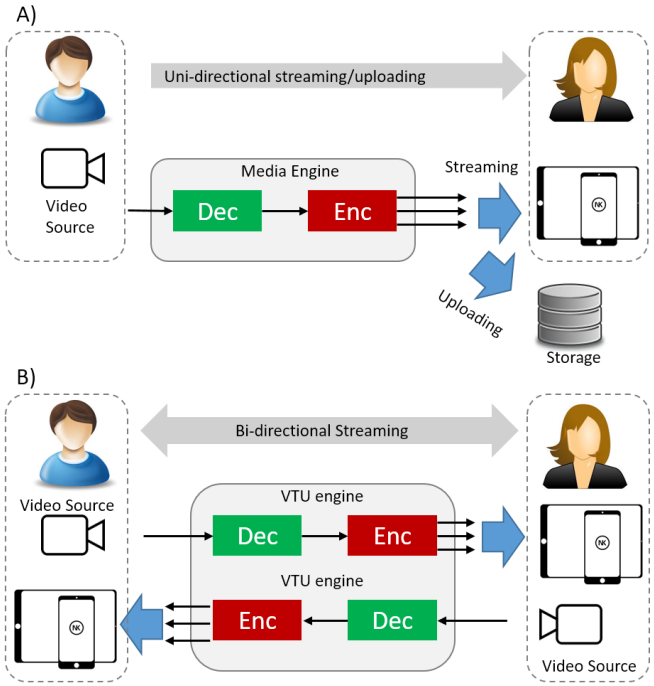

Figure 3. Simplified Media Engine Model A) Unidirectional; B) Bi-directional

To further benefit of HW acceleration, the AVConv tool running in the VTU has been modified so that VP8 video encoding tasks [21] can also exploit the resources eventually offered by off-the-shelf GPUs present in the system. In particular, a proprietary GPU accelerated VP8 video encoder can be used to this end [16][17]. The VTU server can also make use of other software tools, different from the Libav library. For instance, the use of the alternative open source library FFMPEG in place of Libav is not only possible, but also very simple and straightforward.

The VTU ME can support a large set of video codecs, and in particular the most recent and popular ones, such as H.264, H.265 and VP8/VP9 [16][17].

In traditional content delivery systems, the users can receive the video streams they have selected. Conversely, the VTU, through the VTU ME, also gives the users the possibility to originate video contents, and share them within a group, either in real time or as recorded video files in a common storage area.

To implement low-latency, real-time video sharing among the users, two different types of streaming functions have been developed, besides the transcoding function.

The first streaming mode is used to transmitting real-time or pre-recorded video contents from a user to the VTU. This way, an originating user can send contents in real time to the VTU, by means of the Real Time Streaming Protocol (RTSP) [22].

By the second VTU streaming mode, indicated as “broadcasting mode”, the VTU can forward any received content to any user in a group. We will say that the video

stream sent by the originating user is published by the VTU to the group.

In this case, each user receives a notification from the ME that a video stream is being published. If interested, the receiving user can thus decide to access the originated stream in real-time. For the sake of generality, the originating user can also decide to publish its video stream to all the users registered at the VTU. Thus, the video stream becomes public, and not limited within a group. Video streams published by the users are recorded by the VTU and saved in a distributed storage system, to be shared at a later time.

The broadcasting mode can use different protocols to transmit real-time video contents from the VTU to the users. Besides RTSP, also the Real Time Messaging Protocol (RTMP) and the HTTP Live Streaming (HLS) protocol are available [23][24].

In broadcasting mode, the source originating the video content can also be a video file, stored in the VTU storage system. This way, pre-recorded video contents can be streamed to single users, to an entire group or to all the users. This option can be used, for instance, by event organizers to distribute pre-defined video contents to all the participants to the event.

To facilitate the sharing of pre-recorded video contents among users, the VTU can also provide storage capabilities. Typically, video contents originated by the users are shared through cloud-based applications, which require the transmission of the video file from the user's wireless device to a centralized cloud infrastructure through the core network. Any other interested user can retrieve the video file of interest, by downloading it through the core network. However, during a crowded event, such operations are usually significantly slow or even impossible, due to the high density of users in a relatively small area, who rapidly saturate the backhaul connection to the core network. Preliminary experiments in typical crowded events have shown that the upload of a file of size equal to 100MB at peak hours can take times of the order of tens of minutes. Conversely, the same upload to the local VTU storage system, though accessed through the same WIFI network, can take no longer than 10 to 20 seconds.

Thus, the VTU offers local storage capabilities to groups of users. This way, all the members of a group can locally store or retrieve contents to the group storage area, without accessing the core network through the backhaul connection. In addition, the local storage can be used by event organizers to stream video contents to all the users.

The VTU ME offers a web based interface, dedicated to management and monitoring tasks. Such an interface is available only to system managers, and is accessed through an ad hoc web interface. To monitor the VTU status and performance during service, several metrics are generated and constantly sent at regular and modifiable basis to a user defined network address. Currently, monitoring data generated by the VTU is redirected to a local instance of InfluxDB database, a popular open source database solution, oriented to collecting and managing time-series data [25]; then, such records are graphically arranged and visualized into a browser window by Grafana, a popular dashboard for displaying time series metrics [26]. The collected metrics are related to CPU, GPU (when available), memory, and disk usage; also, encoding performances are captured and visualized, such as the number of transcoded frames per second.

2) The VTU UGDM

The VTU UGDM component collects and continuously monitors relevant information about participants and groups during a crowded event, and stores it into an ad hoc database, the User and Group Database. Any user connecting for the first time to the network during a crowded event can download the VTU App (described in the following sub-section), and register to the system. This way, users are recorded to the VTU framework, and are permanently identified by a suitable randomly generated key, that remains unchanged for the entire duration of the event, combined with the user name and/or nickname. Through the App and interacting with the UGDM, users can then create a group, to share video contents. For each user, the database stores all the relevant information about identity, group belonging, connectivity, and Service Level Agreement, so as to provide to all the users the required type of service. The UGDM component also provides basic security functionalities, such as, for instance, user password management.

B. VTU App

The VTU provides an overall framework that enables IVSs in a simple and straightforward way. Two different types of interfaces are available to the users, to benefit of the services made available by the VTU.

The first interface is based on web services, and is accessible by any browser. However, to improve user experience, a specific VTU App has been developed.

Through the App, the users can interact with the framework. In particular, they can register to the VTU system, as described above. Once registered, the users can create groups, for sharing video contents.

To each group, a shared storage area is assigned, where all the users of the group can read or write video contents. This storage area is private and dedicated to the group. Once the group is created, the users belonging to a specific group can benefit of the functionalities made available by the VTU.

The simplest services accessible through the App include video file exchange and video chatting. In addition, other services can be used, not specifically linked to video functionalities. For instance, geo-location and/or navigation functions can be presently provided combined with augmented-reality-based applications. Though, for the sake of brevity, this type of services will be not further discussed in the following, not being related to video.

C. VTU connectivity

To provide IVSs now, anywhere and with any device, two features play a fundamental role from the connectivity point of view, i.e., bandwidth, and latency, both on the user and on the control plane.

The 5G architecture will provide significant advances related to such parameters. In particular, among the goals of the new 5G network, some are of specific interest for IVSs

and crowded events. In fact, the 5G network will enable eMBB (enhanced Mobile Broad Band) types of services, and will consider scenarios with high user device densities (more than 10000 per km square), and low latency. In particular, in the 5G use cases, three latency ranges are considered: high (greater than 50 ms), medium (10-50 ms) and low (1-10ms) [3].

However, the 5G network will start deployment and operations after 2020. Thus, other scenarios must be considered to provide now IVSs. To this end, one solution presently consists in using a WIFI access network to provide the needed connectivity between the VTU App and the VTU VNF. The second option is the use of the CESCs, as discussed in Section III. In this case, the present 4G mobile network architecture can be used, in conjunction with the CESC. A specific solution that can be used to deploy the VTU in a 4G scenario can be found in [10]. Moreover, standardization aspects related to the use of the VTU in 4G and 5G scenario are discussed in [7].

V. The Pervasive Video Use Case

A possible IVS use case can be described by the following two scenarios.

"You're at a stadium, where a football match takes place. Your team scores a goal but you are not in the best position to appreciate it or the action was confusing and you did not realize who scored the goal and how. You would like to have the possibility to watch on your smartphone the most relevant actions from different points of views"

Or:

"You are attending a crowded concert in the front row close to the stage and you want to show to other friends attending the concert far from the stage some videos in real time, picturing the performance in progress. Also, the concert organizers could decide to show on the gigantic main screen a collage of real time videos coming from spectators to give them a more immersive and engaging experience."

In this type of contexts, there is an overwhelming demand for services that give the possibility to the users to have videos on their smartphones or tablets on demand, as services provided, for instance, by the Stadium. Also, the innovative aspect with respect to the traditional Video Content Scenario is that the users are not only the consumers of video contents, but also the generators, with the will to share with other users their video contents.

From the technological perspective, what is needed to implement such type of services is a networking infrastructure featuring a very rapid upload and download of large files, such as HD videos. In addition to that, the possibility to process in a highly effective way video streams is a mandatory function to provide enhanced services. In particular, the capability to adapt the video format to the one required by the users' devices is one of the critical functions needed for this type of service. In fact, video format adaptation, usually referred to as video transcoding, is a compute intensive workload, in particular with high definition video. The VTU, implemented in a MEC environment, thus represents a possible answer to that demand coming from the market. The HW-accelerated transcoding of video streams can help in reducing the computational workload of mobile terminals converting video streams from the uploaded format to one more suitable for the receiving terminal, increasing its battery life.

The whole process of upload, transcoding and download takes place locally in the MEC server (Fig. 4) offloading the backhaul connection towards the core network. This reduces latency and avoids backhaul traffic congestion.

To this end, the MEC server must be equipped with its own high performance storage where all the videos uploaded from the users are kept for a certain amount of time, for instance a week. During this period a suitable application can make them available on demand outside the perimeter of the stadium, e.g., at home. The spectators during a sporting event can then upload many videos and delete them immediately to preserve memory space on their mobile devices, having the possibility to choose at a later time which one to download.

To provide these services, the Stadium or the event organization will then only need to make available the VTU App, to be downloaded on spectators' smartphones.

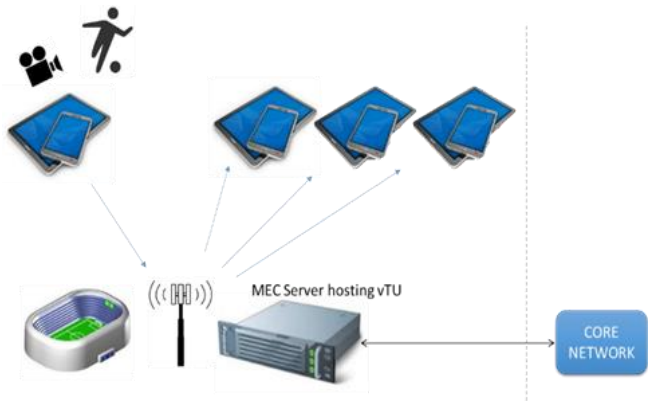

Figure 4. VTU use case: providing low lantency video sevices during localized crowded events leveraging MEC architecture.

VI. VTU And Hardware Acceleration

Although software-only functions can give acceptable performance in many applications, when compute-intensive workloads running at the data plane are of interest, such as those based on video data processing, quite poor results can be obtained. In these cases, to reach the expected performance, it is often necessary to consider a slightly different approach that involves the use of Hardware accelerators. In general, managing HW accelerators and making them transparently available to every VNF goes against the assumption of every virtualized environment, of having a uniform HW platform made of CPU-only computing elements. The presence of HW accelerators bound to a virtual function implies the use of a SW layer that must be HW-aware, thus significantly complicating system management operations and scalability [5][14]. Though, the advantages of HW acceleration can be so preponderant, in particular when performance, latency or Service Level

Agreement (SLA) requirements are challenging, so that not considering them can push a commercial product out of the market. In fact, acceleration is not just related to performance, but also to the reduction of the number of physical servers, footprint, network appliances and power consumption. In short, it can make the difference in the commercial proposition of a product.

A distinctive feature of the VTU is the possibility not only to run on general purpose CPUs but also to exploit the Hardware acceleration provided by a GPU, to improve the compute performance of video codecs. To this end, two different architectural approaches can be used. The first one, also known as "cooperative CPU- GPU" makes use of a GPU to offload the most compute-intensive functions of the video codec (usually, the Motion Estimation block), while the main algorithm is kept running on the CPU. The second approach, conversely, uses full HW implementation of video codecs. Today, various HW versions of the most popular encoding schemes, such as H.264, HEVC, VP8 and VP9, are available [20][21]. The fully HW approach can provide higher compute performance than cooperative CPU-GPU algorithms. However, the HW approach very often lacks the flexibility in service management needed by service operators, thus the cooperative approach is still preferred in many real-life implementations. The VTU can adopt both GPU-accelerated approaches. In fact, it can use the Nvidia NVEnc encoder for the H.264 and H.265 encoding schemes [20]. Also, the CPU-GPU cooperative approach described in [16][17] can be used for the Google open Source VP8 encoder.

VII. VTU Performance

Many tests were carried out to achieve a full performance characterization of the VTU, both for the SW-only version, and the GPU-accelerated one.

The SW-only version of VTU was tested on two different micro servers, based on ARM and INTEL X86 CPUs respectively. The ARM-based micro-server is a NXP commercial evaluation board equipped with a LS2085A processor (with 8x A57 cores @1.8 GHz) and 16GB DDR4 system memory. The INTEL-based server is a commercial platform (GOMA FlexPAC Industrial portable workstation) equipped with an Intel Xeon E5-2630v3 2.4GHz, 8 Core CPU and 64 GB DDR4 RAM.

The GPU-accelerated version of VTU was tested on the same GOMA server, this time equipped with one NVIDIA Quadro M4000 GPU. During all the tests it was verified that the System Memory (RAM) was not completely used, to be sure that the results were not influenced by the different quantity of RAM installed in Intel-based, rather than ARM-based micro-servers.

In the following, only some meaningful results are presented and discussed, for the sake of brevity. In all described tests, the same H.264 Full HD video file (1080x1920 resolution) was used as input. In particular, Fig. 5 shows the results obtained with the VTU featuring the H.264 transcoding (expressed in frames per second) without HW acceleration (SW-only) and with HW acceleration (using a GPU). The processing implies decoding from the input format to the one required as output.

The VTU provided four different video resolutions as output, in four different transcoding tests: VGA (480x640 pixel), HD480 (480x852 pixel), HD720 (720x1280 pixel), HD1080 (1080x1920 pixel). The vertical axis represents the output resolution, while the horizontal axis indicates the achieved output frame-rate in frames per second (fps). The Encoder used in SW-only VTU for H.264 is X264. The Encoder used by the GPU for H.264 and H.265 is NVIDIA NVENC.

Fig. 5 collects the results of the tests achieved with H.264 encoders. Only one session was launched for each test. As one can easily see, the performance for the three HW platforms are very different. In particular, the ARM performance is very poor, to the point that it is not conceivable a utilization in a real-time scenario. The improvement achieved by using the GPU compared to a SW-only solution is remarkable in all cases. This confirms the need of GPU acceleration especially in modern and future scenarios where even higher video resolutions are going to be used. Another important aspect to emphasize is related to the occupation of compute resources during transcoding. Although in SW-only mode CPU resources were completely occupied, (all the CPU cores were running at 100%), using the GPU, both CPU and GPU resources were only partially used. This fact led to a second set of tests in which the multi-session performance was analyzed. In this new set of tests, the focus was on a single case, i.e., H.264 HD1080, launching 2, 4, 8, and 16 concurrent transcoding sessions. The results are reported in Fig. 6 and Fig. 7.

Considering the SW-only implementation (Fig. 6), the performance of each single session decreases with the total number of executing sessions. Again, the ARM performance appears very low, being almost a fourth of the Intel one.

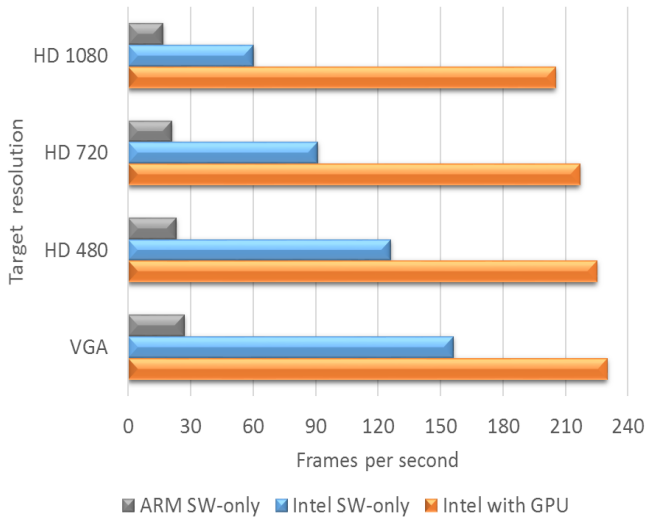

Figure 5. H.264 single session encoding performance (higher is better) measured on three different HW platforms, for different output resolutions.

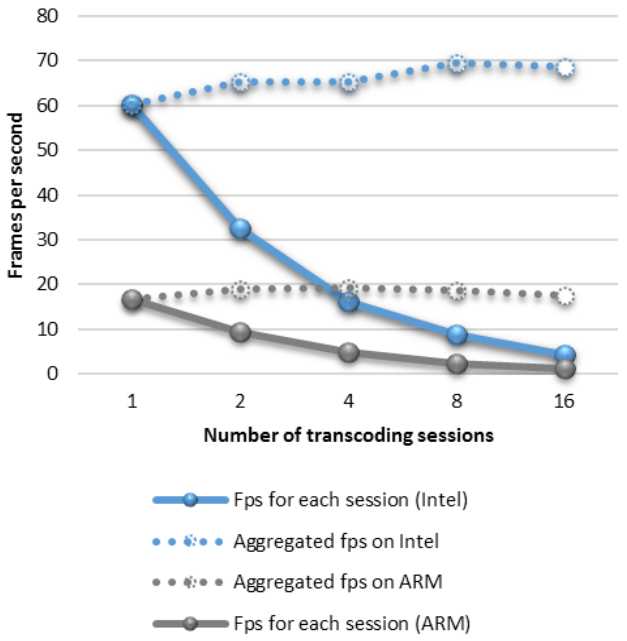

Figure 6. H.264 HD1080 encoding, SW-only, in multi-session transcoding tests (higher is better). Blue lines refer to INTEL, grey to ARM. Performance refers to each single session (solid line) and to aggregated sessions (dotted lines).

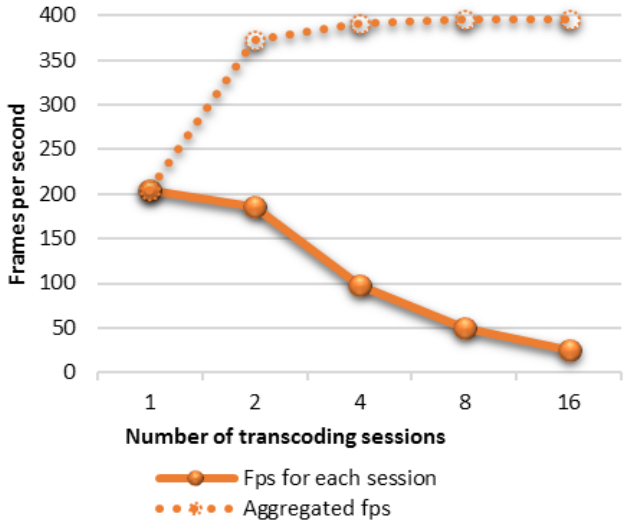

Figure 7. H.264 HD1080 encoding, using a GPU, in multi-session transcoding tests (higher is better). Performance refers to each single session (solid line) and to aggregated sessions (dotted lines).

Comparing the global performance of 1 session to that with 2-4-8-16 concurrent sessions (aggregated fps) the result is very similar as you can see from the dotted lines. This can be easily justified considering that the CPU occupation during the processing is always around 100% also running a single session. The same is not true using the GPU (Fig. 7). In this case the CPU is only partially used because the workload is mainly offloaded to the GPU whose resources are, in turn not fully used (Fig. 8). Using the GPU with 16 concurrent sessions we reach 24.75 fps for each session, for a total of 16x24.75=396 aggregated fps. The 396/69 ratio brings to a 5.7x gain in performance using the GPU respect to a SW-only solution. Furthermore, during the GPU test with 16 transcoding sessions the CPU was running at 70% giving it the possibility to run other tasks. This was not possible with SW-only solution, because in such a case the CPU was always 100% occupied.

It is interesting to analyze what are the power figures of the three HW platforms used to run the tests (multi-session performance). Fig. 9 shows the power consumption, expressed in Watt.

The measures have been carried out in DC, testing the current flowing on the reference voltages of the motherboards (12V, 5V. 3.3V), to the end of excluding the contribution of the main AC-DC power supply and having a more comparable setup between platforms. We used a current clump with a resolution of 0.1A, resulting in a maximum uncertainty of 1.2W (on 12V voltage rail).

As expected, the NXP micro-server exhibits the best results, starting from 31W in idle state and going to 48W when running the VTU application. It is worth noting that 9.5W of the 31W are dedicated to the fans that always run at maximum speed; if the NXP micro server could properly control the fan speed, its power consumption would improve significantly. Regarding the behavior of the Intel platform, with and without GPU, the results could appear unexpected because there are conditions in which the presence of the GPU does not increase the total power consumption, but rather decreases it. This can be explained considering Fig. 8, which shows the percentage of CPU and GPU resources occupied during transcoding (dotted lines). Let us consider, for example, the situation with one session. Intel platform consumes 119W without GPU, and 95W with GPU. However, while w/o GPU the CPU is running at 100% (not reported in the figures), with GPU transcoding requires only 20% of CPU and GPU resources (as in Fig. 8). This is the reason why the power consumption with GPU is in some case even better (lower) than the one w/o GPU.

Comparing the efficiency of the two solutions in term of performance/watt (Fig. 10), we see another important advantage of using the GPU. In fact, in the case of 16 concurrent sessions (H.264-HD1080), the gain in efficiency using Intel + GPU is 5.4x compared to Intel (SW-only). This could be, in some way, expected. Less obvious is the greater efficiency of Intel with respect to ARM (for this particular application).

Finally, Fig. 11 shows the performance of the VTU when the transcoding is made starting from the same input file used so far, but converting it to a H.265 format. This transcoding operation is significantly more complex from the computational point of view than the previous one (H.264), as one can see from the reduced performance respect to Fig. 5. We did not report the ARM performance, because it is too low to be considered. Intel performance is very low too, and

the reduction in performance (and efficiency) with respect to the GPU reaches 10x.

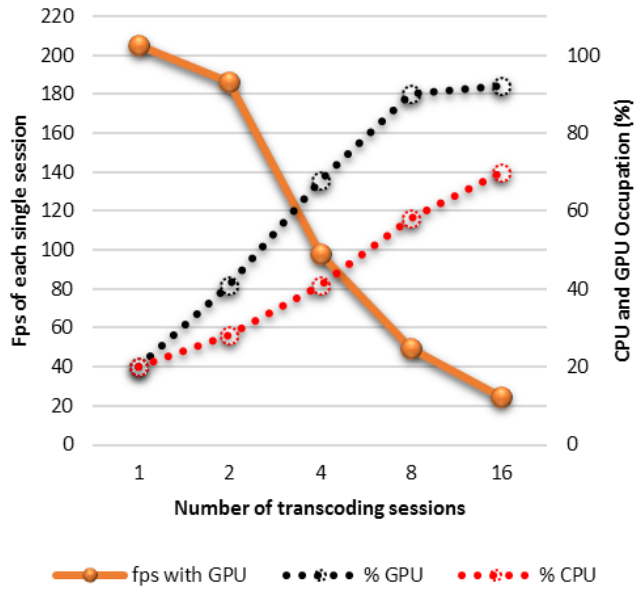

Figure 8. H.264 HD1080 encoding with GPU in multi-session transcoding tests (performance related to each single session) with percentage of CPU and GPU resources utilization.

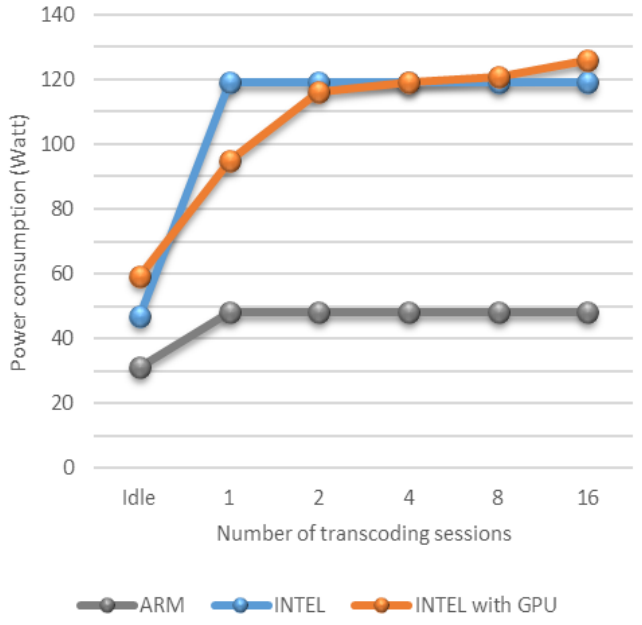

Figure 9. Power consumption (lower is better) of the three HW platforms running the H.264 HD1080 encoding multi-session transcoding tests.

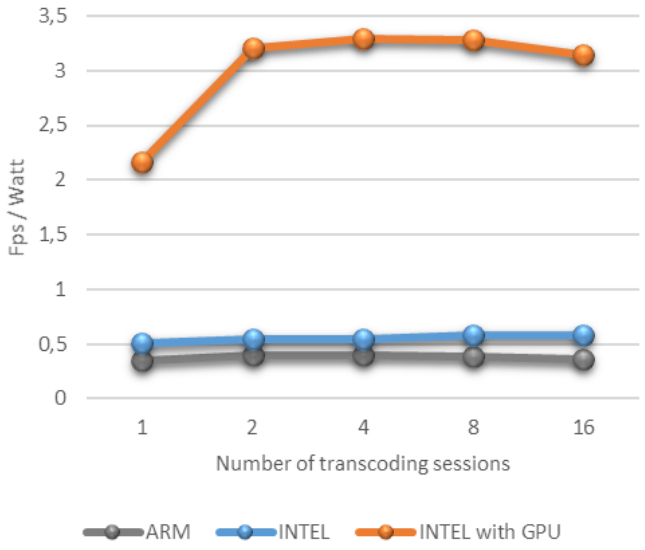

Figure 10. Efficiency of the three HW platforms (expressed in performance/power) for H.264 HD1080 encoding in multi-session transcoding tests (higher is better).

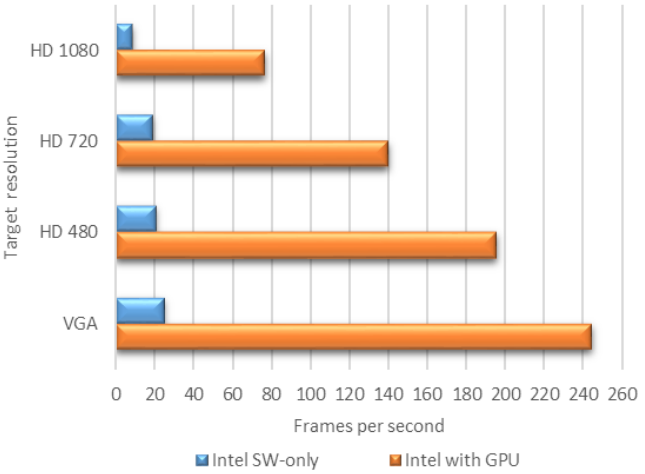

Figure 11. H.265 single session encoding performance (higher is better) measured on two different HW platforms, for different output resolution.

VIII. CONCLUSION

This paper has presented the Video Transcoding Unit (VTU) application, which, leveraging MEC principles, brings high performance video data processing functionalities to the network edge, greatly improving User Experience with mobile terminals. The VTU can be implemented as a SW-only VNF, or be accelerated by a GPU. Specific tests have been reported, showing the clear superiority of the HW-accelerated implementation.

A possible use case has been presented in which the VTU is used in a Stadium or in large public venues during a crowded event like a sporting match or a concert.

In future, we will promote further development of this platform, using it for different video services to deploy at the edge, such as video analytics or augmented reality. In this context, special focus will be on scalability of computing resources, to provide multi-GPU systems for massive, real-time video transcoding.

ACKNOWLEDGMENT

This research received funding from the European Union H2020 Research and Innovation Action under Grant Agreement No.671596 (SESAME project).

The authors are grateful to Mr. Marco Beccari and Mr. Luca Di Muzio who carried out the laboratory tests described in this paper.

REFERENCES

[1] A. Albanese, P. S. Crosta, C. Meani, and P. Paglierani, "GPU-accelerated Video Transcoding Unit for Multi-access Edge Computing Scenarios," SOFTNETWORKING 2017, April 23-27, 2017, Venice, Italy.

[2] CISCO Corporation. The Zettabyte Era: Trends and Analysis. 2017. [Online]. Available from: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html (accessed 13 Nov. 2017).

[3] 5G Infrastructure Public Private Partnership (PPP): The next generation of communication networks will be Made in EU. Digital agenda for Europe. Technical Report, European Commission. February 2014.

[4] NGMN: 5G White paper (2015). [Online]. Available at: https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf (accessed 13 Nov. 2017).

[5] ETSI: ETSI GS NFV-MAN 001 v1.1.1: Network Functions Virtualisation (NFV); Management and Orchestration (2014)

[6] ETSI: Mobile-Edge Computing - Introductory Technical White Paper (2014).

[7] B. Blanco et al., "Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN," Computer Standards & Interfaces, Available online 4 January 2017, ISSN 0920-5489.

[8] White paper. "The SESAME approach for clustered Small Cell deployments: Introducing advanced coordination and service capabilities through a distributed edge data centre," July 2016. [Online]. Available at: http://www.sesame-h2020-5g-ppp.eu/Dissemination.aspx (accessed 13 Nov. 2017).

[9] White paper. "SESAME: An innovative multi-operator enabled Small Cell based infrastructure that integrates a virtualised execution platform for deploying Virtual Network Functions," July 2017. [Online]. Available at: http://www.sesame-h2020-5g-ppp.eu/Dissemination.aspx (accessed 13 Nov. 2017).

[10] Fajardo et al., "Introducing Mobile Edge Computing Capabilities through Distributed 5G Cloud Enabled Small Cells," Mobile Netw Appl (2016) 21: 564. [Online]. Available at: https://doi.org/10.1007/s11036-016-0752-2 (accessed 13 Nov. 2017).

[11] Z. H. Chang, B. F. Jong, W. J. Wong, and M. L. D. Wong, "Distributed Video Transcoding on a Heterogeneous Computing Platform," APCCAS, 25-28 Oct., 2016, Jeju, South Korea.

[12] S. Li et al., "QoE analysis of NFV-based mobile edge computing video application," in 2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC).

[13] S. Fu, J. Liu, and W. Zhu, "Multimedia Content Delivery with Network Function Virtualization: The Energy Perspective," 2017 IEEE MultiMedia.

[14] P. Paglierani, "High Performance Computing and Network Function Virtualization: A major challenge towards network programmability," 2015 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Constanta, 2015, pp. 137-141.

[15] P. Paglierani et al., "Techniques for providing Software and Hardware Acceleration to VNFs running on the Edge Cloud," EUCNC2017, June 12-15, 2017, Oulu, Finland.

[16] P. Comi et al., "Hardware-accelerated high-resolution video coding in Virtual Network Functions," 2016 European Conference on Networks and Communications (EuCNC), Athens, 2016, pp. 32-36.

[17] P. Paglierani, G. Grossi, F. Pedersini, and A. Petrini, "GPU-based VP8 encoding: Performance in native and virtualized environments," 2016 International Conference on Telecommunications and Multimedia (TEMU), Heraklion, 2016, pp. 1-5.

[18] SESAME D3.1 "CESC Prototype design specifications and initial studies on Self-X and virtualization aspects," June 2016. [Online]. Available at: http://www.sesame-h2020-5g-ppp.eu/Deliverables.aspx (accessed 13 Nov. 2017).

[19] Libav. [Online]. Available at: http://libav.org/documentation/ (accessed 13 Nov. 2017).

[20] NVIDIA NVENC Programming Guide [Online]. Available at: https://developer.nvidia.com/nvenc-programming-guide (accessed 13 Nov. 2017).

[21] WebM Video Hardware RTLs [Online]. Available at: https://www.webmproject.org/hardware/ (accessed 13 Nov. 2017).

[22] IETF RFC7826, "Real-Time Streaming Protocol Version 2.0," [Online]. Available at: https://tools.ietf.org/html/rfc7826 (accessed 13 Nov. 2017).

[23] "Real-Time Messaging Protocol (RTMP) specification," [Online]. Available at: http://www.adobe.com/devnet/rtmp.html (accessed 13 Nov. 2017).

[24] "HTTP Live Streaming (HLS)," [Online]. Available at: https://developer.apple.com/streaming (accessed 13 Nov. 2017).

[25] InfluxData InfluxDB project. [Online]. Available at: http://docs.influxdata.com/influxdb/v1.3/ (accessed 13 Nov. 2017).

[26] Grafana project. [Online]. Available at: http://grafana.org (accessed 13 Nov. 2017).