Adaptive 360° Video Streaming over a Federated 6G Network: Experimenting In-Network Computing for Enhanced User Experience

Andrea Caruso† andrea.caruso@phd.unict.it
Giovanni Schembra† giovanni.schembra@unict.it
Christian Grasso† christian.grasso@unict.it
Juan Brenes* j.brenes@nextworks.it
Pietro G. Giardina* p.giardina@nextworks.it

Giada Landi* g.landi@nextworks.it
Leonardo Lossi* l.lossi@nextworks.it
Gabriele Scivoletto* g.scivoletto@nextworks.it

*Research & Development, Nextworks, Pisa, Italy
†University of Catania, CNIT Research Unit, Catania, Italy

***Abstract*—The entertainment and gaming industries continue to evolve, so they are becoming increasingly reliant on advanced network capabilities to deliver immersive, real-time experiences. In-network computing (INC) is a transformative paradigm in the design of the network architecture in 6G networks. It facilitates the offloading of computational tasks from devices to edge nodes and central servers, enabling faster data processing in network nodes and reducing latency and response times for high-demand applications, enhancing efficiency. This way, it will be possible to provide the final users with advanced applications like Augmented Reality (AR) and Virtual Reality (VR). In this paper we introduce a 6G Network Infrastructure testbed designed for application execution and testing, supporting INC, and we present an Adaptive 360° Video Streaming service using INC facilities. The testbed has been designed to represent a vertical application for SLICES-RI and SUNRISE-6G projects. Some numerical results will assesses the performance of this dynamic video streaming system.**

***Index Terms*—6G, INC, Augmented Reality, Virtual Reality, Adaptive video-streaming**

I. Introduction

The advent of 6G networks represents a paradigm shift in the telecommunications landscape, guaranteeing unprecedented levels of speed, reliability, and ultra-low latency [1], changes that partly started with 5G networks [2]. Starting from 5G networks, the network infrastructure will evolve as a "network of networks", giving great importance to the presence of private 6G networks, which will be integrated with public networks with the task of providing unprecedented data speeds, ultra-low latency and highly reliable communications. This transformation is achieved thanks to technological innovations, and in-network computing (INC) plays a significant role in enabling these advancements. INC involves the integration of data processing directly within the network infrastructure, addressing the limitations of traditional cloud-centric approach by significantly reducing latency and enhancing overall network efficiency [3].

This approach is in line with the central idea of 6G, which is to realize a network architecture that is no longer characterized by a centrality, but is distributed, in which computing resources are increasingly decentralized, approaching the edge of the network to meet the needs of latency-sensitive applications. Some previous work has been conducted focusing on moving computing to the edge of the network in scenarios such as Flying Ad-hoc NETwork (FANET) [4] and Vehicular Ad-hoc NETwork (VANET) [5], with resource optimization techniques deployed at the edge that are certainly inferior to their remote cloud counterparts. Among the most demanding applications in terms of low latency, we cannot fail to mention those related to augmented reality (AR) and virtual reality (VR) [6]. The entertainment and gaming industries are undergoing a significant transformation, driven by rapid advancements in technology and increasing consumer demand for immersive, interactive experiences. As these industries evolve, they require robust and highly responsive network infrastructures to support next-generation applications. Indeed, AR and VR applications impose stringent requirements on network performance, especially in terms of latency, bandwidth, and reliability. If in VR games minimal delays can cause disorientation and user dissatisfaction, when we move to the field of AR-assisted medical procedures, we understand how a robust network infrastructure that guarantees ultra-low latency is of fundamental importance.

In this context, INC plays a crucial role for AR and VR scenarios in 6G networks. By performing data processing at the edge nodes or even directly on network devices, INC dramatically reduces the distance that data has to travel, resulting in lower latency. Furthermore, INC improves the ability of 6G networks to handle the massive amounts of data associated with real-time environment rendering, user interaction, and immersive content, which will increase exponentially in the coming years. By distributing processing across the network, it is possible to ensure that even as data volumes increase, performance remains robust and scalable. In addition to latency reduction and scalability, INC also contributes to the resilience

978-3-903176-66-9 ©2024 IFIP

and flexibility of 6G networks [7]. By enabling localized data processing and decision-making, networks can better adapt to dynamic conditions and offer continuous service even in the face of localized failures or congestion. This is particularly important for AR and VR applications.

Data from AR/VR applications arriving at the edge must be processed in order to be able to return a result to the end user. For this reason, the implementation and positioning of Virtual Functions (VFs) at the edge of the network, capable of working with lower computational resources than what is possible at the remote cloud level [8], is another aspect to take into account. In many cases, depending on the performance of the network itself, being able to reduce the actual amount of data that passes through it can make the difference.

The work presented in this paper falls within this context, which concerns the implementation of the Adaptive 360° Video Streaming Closed-loop Encoding VNF designed in an early stage by some of the Authors of this paper in [9]. This VNF is deployed and executed in a Federated 6G Network Infrastructure based on the INC paradigm for validation purposes. This VNF is capable of implementing hierarchical compression, considering the user's viewport determined by the user's movements with a headset visor. The testbed is composed of various orchestration components for the 6G Core part, the Edge Computing part and the Software Defined Network (SDN) part, capable of interacting with the physical devices connected to the testbed itself. The testbed has been designed to represent a challenging vertical inside the SLICES-RI [10] and the SUNRISE-6G [11] projects.

The rest of the paper is structured as follows. Section II describes the vertical application regarding a 360° video streaming service which has been tested and validated, while in Section III there is a description of the implemented testbed. Section IV shows the setup of the experiments, while Section V presents some measurement results. Finally, Section VI draws some conclusions.

II. Vertical Application Scenario

In this section we will present the Adaptive 360° Video Streaming Closed-loop Encoding (*360° Enc*) VNF, that is able to implement a hierarchical compression of the frames of a video coming from a 360° camera. The goal of this VNF is to divide the frame into zones, each characterized by a certain number of tiles, and apply a different level of encoding to each zone, depending on what the user is seeing with the viewer at that moment. In particular, the zone where the user is focusing his gaze will be encoded with a low level of compression, in order to give a high-quality vision to the user himself, while the other zones will undergo higher levels of compression as they move away from the part that the user is observing.

This setup enables users to experience an immersive 360° scene, allowing them to fully engage with the environment.

On the video-source side, there is a *360° Adaptive Video Encoder* , which handles the conversion of the spherical video coming from the 360° camera into a 2D format. This process involves stitching the spherical image into a flat, planar representation and then encoding it for transmission. The most common approach is to use an equirectangular projection, which maps the 360° spherical image onto a 2D plane. However, alternative projection techniques such as cubic or pyramid projections [12] can also be utilized. The output of this encoder is sent to the *360° Enc* VNF, that is responsible for compressing the video stream. This compression is achieved using a tile-based method [13]–[15], which divides the video into multiple segments or tiles. This VNF is deployed in the closest network access point (AP) of the video-source side, to reduce the latency of 2D video input to the network.

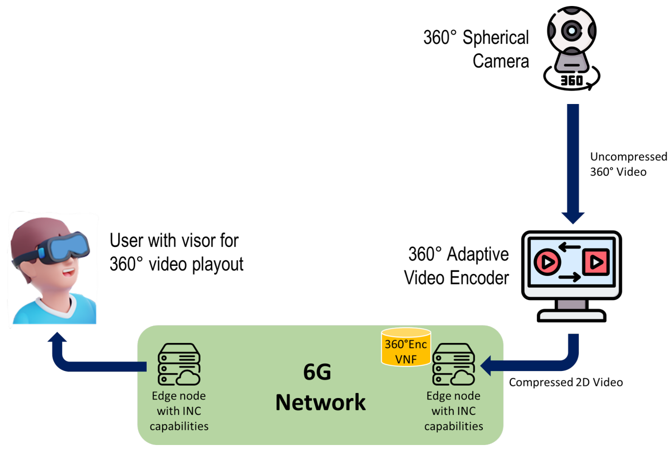

Fig. 1. System Global Architecture

On the user side, the user is equipped with a headset visor. This visor not only displays the video but also tracks the user's head movements, capturing the user's current viewport. This viewport data is then sent back to the *360° Enc* by means of the MQTT protocol, which relays the viewport information to the encoder.

The *360° Enc* VNF utilizes a dynamic compression algorithm that assigns different bitrates to each tile based on its proximity to the user's viewport. The entire video frame is divided into a grid, with $(2R + 1)$ rows and $(2C + 1)$ columns, as shown in Fig. 2. Let I represent the set of all tiles, defined by their coordinates. This set is further divided into four distinct zones, Z_i for $i \in \{1, 2, 3, 4\}$, such that:

$$\begin{cases} Z_i \cap Z_j = \varnothing, \forall i, j \in \{1, 2, 3, 4\} \\ \bigcup_{i=1}^{4} Z_i = I \\ Z_i \text{ is geometrically contained within } Z_{i+1}, \forall i \in \{1, 2, 3\} \end{cases} \quad (1)$$

Consider a matrix of tiles, denoted as $\boldsymbol{T}$, similar to the one shown in Fig. 2. Let $T_{[r,c]}$ denote an individual tile located at row r and column c. The set of tiles belonging to each zone z is represented by $\hat{I}^{(z)}$, where $z \in \{1, 2, 3, 4\}$. Zone Z_1 contains the tile at the center of the user's viewport, designated as $(0, 0)$, which is encoded with the highest quality. Zone Z_2 encompasses the area that lies within the user's immediate field of view and is encoded at a good quality. Zone Z_3 includes

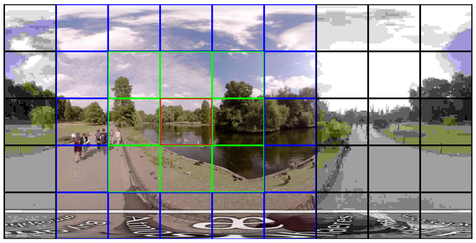

Fig. 2. Video frame divided into four tile zones

areas that are visible when the user moves their eyes, thus requiring a moderate encoding quality. Finally, Zone Z_4 covers regions that are outside the user's direct field of view, only visible with head movement, and can therefore be encoded with the lowest quality.

The viewport of the user is determined by the absolute position of their head relative to a reference point in space. Since the user's head movements are tracked in real-time by the headset, the viewport can shift dynamically. The quality of the video perceived by the user is dependent on how well the video encoding matches the current viewport. To ensure that the *360° Enc* VNF can optimize the encoding quality for each tile in a given frame F_n, the user headset regularly transmits the latest viewport coordinates from the VNF. However, due to network latency, this viewport information might become outdated by the time it reaches the source side. As a result, frame F_n may be encoded based on viewport data that is no longer current. When the encoded frame is transmitted back to the user, additional delays can cause further discrepancies between the encoded content and the actual viewport at the time of display.

In summary, the user's perceived video quality is influenced by the viewport changes during the round-trip time, which in turn affects the set of compression ratios $(\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4)$ applied to the different zones. To quantify the quality experienced by the user, the Peak Signal-to-Noise Ratio (PSNR) is employed. PSNR measures the difference between the original frame before encoding and the frame reproduced by the user headset. The overall PSNR for a frame is determined by calculating the PSNR for each zone, z, and applying a weighting factor η_z to each zone's contribution to the total PSNR.

III. Experimentation Environment

The Adaptive 360° Video Streaming Closed-loop Encoding VNF described in Section II was tested and validated in a Federated 6G Network Infrastructure based on the INC paradigm.

Testing applications and protocols that have to be deployed in 6G network environments require a testbed that is comprehensive and closely mirrors the real-world conditions. This section provides an overview of the two core components of the developed testbed: the first component focuses on the deployment of a 6G private network infrastructure, while the second component describes how this network is integrated with an experimentation platform used in the VITAL-5G project to enable automated experiment deployment and execution.

A. 6G Private Network Infrastructure

The high-level architecture of the testbed is illustrated in Fig. 3, featuring a 6G private network deployed at the University of Catania. The setup includes a Cumucore 5G Core Network incorporating both User Plane Functions (UPF) and Control Plane Functions (CPF). The Radio Access Network (RAN) is emulated using UERANSIM, allowing for the creation of multiple gNBs and UEs. The testbed also incorporates a multi-technology edge platform that supports the deployment of virtual edge applications, following the INC paradigm, whether based on virtual machines (VMs) or containers, using OpenStack and Kubernetes.

On top of the network infrastructure, as depicted in Fig. 3, the testbed includes a management system composed of: (i) an NFV Orchestrator (NFVO) enablign the onboarding and lifecycle management of Virtual Network Functions and Network Services; (ii) A Virtual Infrastructure Manager (VIM) supporting the provisioning of the applications on the edge resources; (iii) A Monitoring Platform. The current setup uses ETSI Open Source MANO (OSM) as NFVO and an OpenStack controller as VIM. The Monitoring Platform, used for measuring the parameters that will be shown in the Section V, is a custom-built solution that allows for the dynamic creation and configuration of data sources at the network, device, and service levels. Specific data sources are configured to implement mechanisms for data retrieval, aggregation, and processing. This platform leverages on the open-source tools Telegraf, Kafka, InfluxDB, Prometheus, and Grafana.

B. 6G Platform Federation for Automated Experiments

To enhance the experimentation and automation capabilities of the testbed described in Section III-A, the 6G private network platform at the University of Catania has been federated with an instance of the VITAL-5G EU-funded project's 5G experimentation platform [16], which is deployed at the Nextworks testbed in Pisa (see Fig. 3). The VITAL-5G Platform is designed to enable and simplify the experimentation of vertical services over mobile network infrastructures with computing capabilities and programmable interfaces. The design is based on the concept of Network Applications (NA) [17], which extend the vertical application concept enabling the specification high-level network characteristics, monitoring metrics for services, computing resources, and integrated devices (e.g., UAVs, IoT Gateways, sensors, etc.). The VITAL-5G Platform [18] offers services for the onboarding and provisioning of NAs and execution of experiments on top of NAs.

As shown in Fig. 3, the three core components of the VITAL-5G platform (the Service Catalogue, Service LCM, and Experiment LCM) have been integrated into this scenario. These components leverage specific plugins to adapt universal

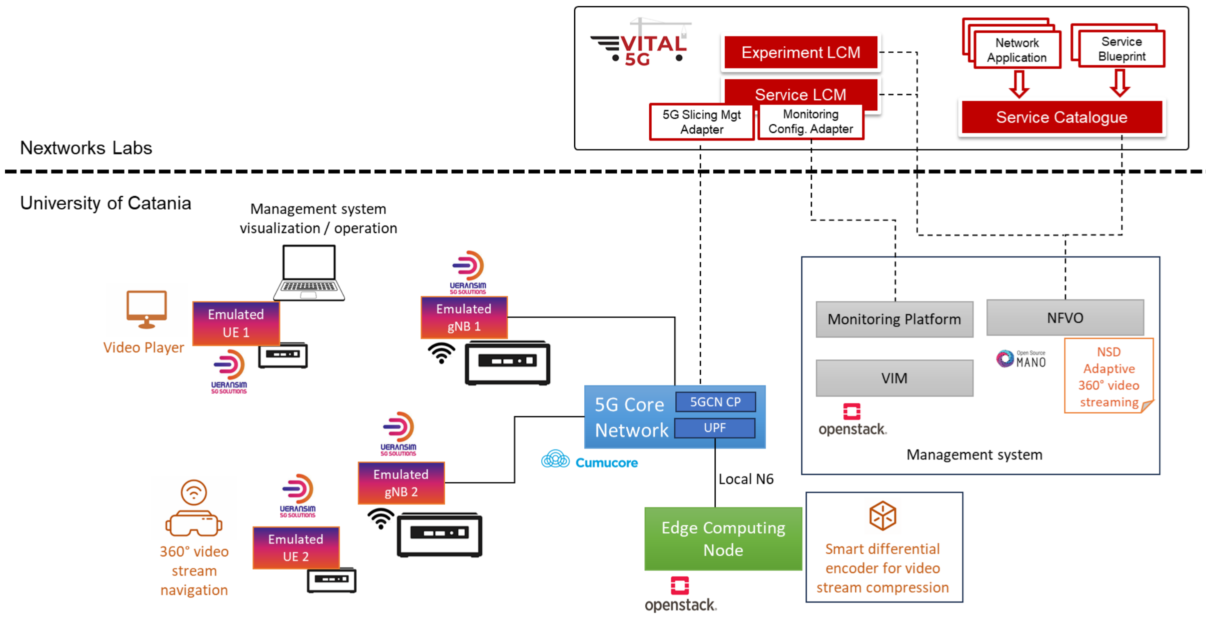

Fig. 3. Testbed platform, infrastructure and services

southbound interfaces of the VITAL-5G Platform to the specific Slice lifecycle management, NFVO and Monitoring APIs of the testbed management system.

The VITAL-5G Service Catalogue interfaces directly with the NFVO of the testbed, in this case, the ETSI OSM, for managing the descriptors associated with vertical applications. The Service LCM is tasked with provisioning, runtime lifecycle management, and the termination of end-to-end services composed of NA . This includes provisioning application components through the NFVO, automated 5G slice management through the Cumucore 5G slice management APIs, and configuration and activation of monitoring jobs through the Monitoring Platform. Finally, the Experiment LCM enables the execution of test scripts, triggering predefined actions on application components using OSM primitives. This facilitates consistent experiment configuration, execution, and reporting, improving the validation and comparison of results across multiple test iterations.

IV. Experiment Setup

This section outlines the setup used to evaluate the Adaptive 360° video streaming service proposed in this paper, where users equipped with VR headsets are connected via a 6G network implementing INC services. The experimental configuration involves an end-to-end service deployment facilitated by a Smart Differential Encoder application, which operates on a VM located at the network edge. This setup is illustrated in Fig. 3. The 360° video stream from the on-site user's VR headset is processed at the edge, with the encoding dynamically adjusted based on network conditions and user movements to optimize the video quality experienced by a remote user.

The entire service is managed by the VITAL-5G Platform, with the adaptive encoder represented as a single NA deployed at the edge. The blueprint for this application specifies an enhanced Mobile Broadband (eMBB) network slice, ensuring a guaranteed uplink data rate for the VR headset's outgoing traffic and a guaranteed downlink rate for the remote user's video player. The monitoring metrics include application-specific parameters like PSNR (Peak Signal-to-Noise Ratio), as well as computing resources (CPU and memory usage) and network statistics. Both the NA and Vertical Service blueprints are stored in the VITAL-5G Service Catalogue. Inside ETSI OSM, the service is defined by a Network Service Descriptor (NSD) consisting of a single VNF, deployed on a VM with 10 CPUs, 16 GB of RAM, and a 50 GB disk. The NSD and the VNF Descriptor (VNFD) are onboarded in the OSM catalogue, while the VM image is managed by the OpenStack controller in the testbed at the University of Catania.

Service provisioning and experiment execution are initiated through the VITAL-5G Platform and managed by the testbed's system. The desired Vertical Service is selected from the VITAL-5G catalogue, and the provisioning request is made via the Service LCM, which specifies the IMSIs associated with the on-field VR headset and the remote user's video player. The VITAL-5G platform automates the creation of the required network slices, deploys the Network Service on ETSI OSM, configures it using day0-day1 primitives, and activates monitoring jobs to collect metrics from the VM. As a result, the network slices are instantiated within the Cumucore 5G Core Network, the VM is launched at the edge node controlled by OpenStack, and the mobile connectivity is successfully established. The total provisioning time is approximately 10 seconds, primarily due to VM creation and application-level configuration. Once the service is provisioned, experiments

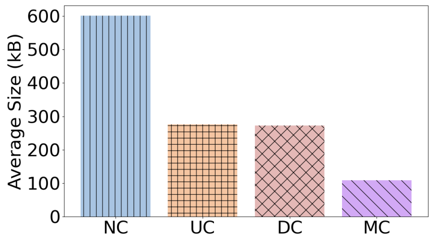

Fig. 4. Average Frame Size

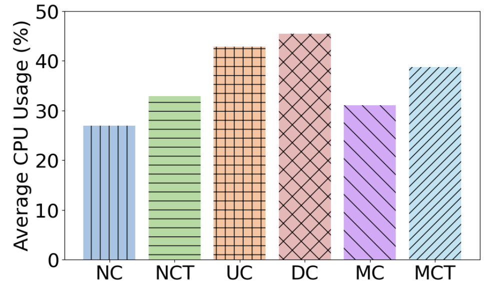

Fig. 5. Average CPU Usage

are conducted through the Experiment LCM on the VITAL-5G platform, where configuration parameters are set, and data collection rounds are initiated.

As depicted in Fig. 3, the performance of the adaptive 360° Video Streaming service under test is evaluated within two distinct coverage areas emulated by two gNBs, with a single User Equipment (UE) connected to each area. One UE is linked to the 360° video source for the field operator, while the other UE is connected to the video player receiving the stream at the remote operator's location.

As said so far, the role of the VNF is to assign a compression ratio to each zone of the video frame. For our evaluation, we utilized the 360° video stream *London Park Ducks and Swans* [19], which consists of 1965 frames at a resolution of 1920x960 pixels, with a frame rate of 30 fps. The VNF was configured to divide the video into a grid of 5x9 tiles.

To simulate realistic user interaction, the video was navigated by a teenager known for quick head movements. Movement data was captured using an Android application we developed, which logs all movements of a smartphone placed in a 360° cardboard-like headset. The application's magnetic orientation sensor tracks rotation angles around the z, x, and y axes (azimuth, pitch, and roll, respectively). These data are collected in the background, allowing the 360° video to play simultaneously while the tracking application runs. The movement data are transmitted from the 360° visor to the *360° Enc* VNF via the MQTT protocol.

To assess the impact of the VNF on VM resources and to estimate the PSNR of the video received by the user, we evaluated six different scenarios: 1) No compression (simple forwarding), 2) No compression (tile-based), 3) Uniform compression (tile-based), 4) Differential compression (tile-based), 5) Max compression (simple forwarding), 6) Max compression (tile-based).

The differential tile-based compression approach applies no compression to zone Z_1 (i.e., $\Gamma_1 = 0\%$), 30% compression to Z_2 ($\Gamma_2 = 30\%$), 70% compression to Z_3 ($\Gamma_3 = 70\%$), and 100% compression to Z_4 ($\Gamma_4 = 100\%$). The compression method used is standard JPEG. For the uniform compression scenario, a uniform JPEG compression with an 81% compression ratio was applied, reducing the frame size to a level comparable to the tile-based differential compression method, as illustrated in Fig. 4 and detailed in Table I.

Each scenario was monitored for 10 minutes to analyze the VM resource usage. To quantify the overall video quality using PSNR, we assigned the following weighting factors to each zone's contribution: $\eta_{z1} = 0.4$, $\eta_{z2} = 0.3$, $\eta_{z3} = 0.2$, $\eta_{z4} = 0.1$ [12], [20].

TABLE I
AVERAGE FRAME SIZE

Scenario	Size
No Compression	601.38 kB
Uniform Compression	275.66 kB
Differential Compression	272.93 kB
Max Compression	109.18 kB

V. EXPERIMENT RESULTS

The experimental results highlight the impact of different compression techniques on CPU and RAM utilization, data throughput, and video quality. The data collected for each metric are analyzed and discussed below.

A. CPU Usage

The average *CPU usage* shows considerable variation depending on the compression method employed, as shown in Fig. 5 and Table II. For the *No Compression (NC)* scenario, the CPU utilization averages 26.93%. This relatively low CPU usage reflects the minimal processing requirements when no compression is applied. In contrast, the *No Compression Tiles (NCT)* method, which applies tiling without compression, results in a slightly higher CPU utilization of 32.87%, suggesting that tiling increases computational overhead. The *Uniform Compression (UC)* method, which applies the same compression rate across all frames, exhibits the highest CPU utilization with an average of 42.89%. This significant increase indicates that the JPEG compression process imposes a substantial computational burden. Similarly, the *Differential Compression (DC)*, which compresses each tile-zone of the frames basing on the user field of view, results in an average CPU utilization of 45.46%. The higher CPU usage here reflects the complexity involved in calculating and applying different compression ratio to each zone. On the other hand, *Max Compression (MC)* achieves a lower average CPU utilization of 31.10% compared

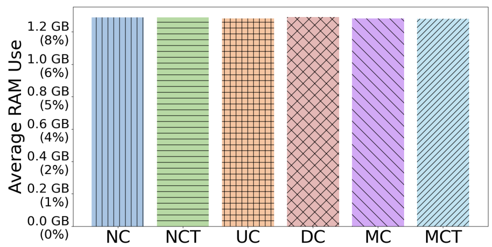

Fig. 6. Average RAM Usage

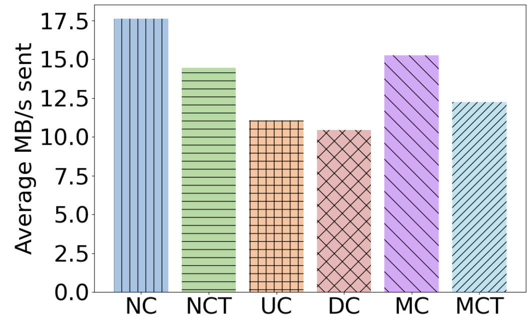

Fig. 7. Average MB sent per second

to uniform and differential compression. Nevertheless, this value is higher than the two no compression scenarios. This suggests that while maximum compression is intensive, it does not match the computational demands of uniform and differential methods. *Max Compression Tiles (MCT)* results in a slightly higher CPU utilization of 38.79% compared to *Max Compression (MC)*, emphasizing that the tiling approach adds an additional computational overhead.

TABLE II
AVERAGE CPU USAGE

Scenario	CPU Usage
No Compression	26.93%
No Compression Tiles	32.86%
Uniform Compression	42.87%
Differential Compression	45.43%
Max Compression	31.09%
Max Compression Tiles	38.75%

B. RAM Usage

In terms of *RAM usage*, the results are notably consistent across the different compression methods (Fig. 6, Table III). For *No Compression (NC)*, the average RAM utilization is 1.29 GB (8.06% of total memory), and this value is identical for *No Compression Tiles (NCT)* and *Differential Compression (DC)*. *Uniform Compression (UC)* and *Max Compression (MC)* both show an average RAM utilization of 1.28 GB (8.02%), indicating that these methods use slightly less memory. The minor variations in RAM utilization across the different techniques suggest that memory usage is not significantly impacted by the type of compression applied. This stability implies that RAM is not a limiting factor in the performance of these compression algorithms.

TABLE III
AVERAGE RAM USAGE

Scenario	RAM Usage
No Compression	1.29 GB (8.06%)
No Compression Tiles	1.29 GB (8.06%)
Uniform Compression	1.28 GB (8.02%)
Differential Compression	1.29 GB (8.06%)
Max Compression	1.28 GB (8.02%)
Max Compression Tiles	1.28 GB (8.02%)

C. Data Throughput

Data throughput, measured in megabytes per second (MB/s), varies according to the compression method used, as it is shown in Fig. 7 and Table IV. The *No Compression (NC)* scenario achieves the highest throughput at 17.62 MB/s, reflecting the lack of overhead associated with processing uncompressed data. *Max Compression (MC)* follows with 15.26 MB/s, demonstrating that while maximum compression reduces throughput compared to no compression, it still performs relatively well. In contrast, *No Compression Tiles (NCT)* and *Max Compression Tiles (MCT)* show reduced throughputs of 14.44 MB/s and 12.24 MB/s, respectively. The decrease is attributable to the additional processing required for tiling, which impacts the overall data transfer rate. The *Uniform Compression (UC)* results in a throughput of 11.07 MB/s, and *Differential Compression (DC)* achieves the lowest throughput at 10.44 MB/s. These reductions reflect the increased processing and transmission overhead associated with these compression techniques.

TABLE IV
AVERAGE MB SENT PER SECOND

Scenario	Average MB/s sent
No Compression	17.62 MB/s
No Compression Tiles	14.44 MB/s
Uniform Compression	11.07 MB/s
Differential Compression	10.44 MB/s
Max Compression	15.26 MB/s
Max Compression Tiles	12.24 MB/s

D. Video Quality (PSNR)

The *Peak Signal-to-Noise Ratio (PSNR)* values provide insight into the video quality achieved by each compression method (Fig. 8, Table V). *No Compression (NC)* yields the highest PSNR value of 49.24 dB, indicating the best video quality with no loss due to compression. In contrast, *Uniform Compression (UC)* results in a PSNR of 31.74 dB, showing a significant decrease in video quality due to the consistent compression applied across all frames to match the *Differential Compression (DC)* frame size, which achieves a PSNR of 37.07 dB, notably better. Finally, *Max Compression (MC)* results in the lowest PSNR of 28.55 dB, reflecting the highest

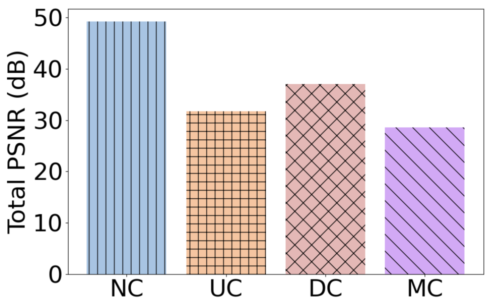

Fig. 8. Average PSNR

level of compression and the resulting substantial loss in video quality.

TABLE V
AVERAGE PSNR

Scenario	PSNR
No Compression	49.24 dB
Uniform Compression	31.74 dB
Differential Compression	37.07 dB
Max Compression	28.55 dB

VI. Conclusions

This study examines the use of Adaptive 360° Video Streaming in a Federated 6G Network, leveraging In-Network Computing (INC). INC significantly enhances streaming performance by offloading computations to edge nodes, thus reducing latency—critical for AR and VR applications where delays impact user experience. The Adaptive 360° Video Streaming service uses viewport-based hierarchical compression to manage network resources more effectively. Differential compression, which adjusts quality based on the user's view, improves video quality but increases CPU usage. RAM usage remains stable, indicating that memory is not a bottleneck. There is a trade-off between compression level and video quality: high compression may lower quality but is suitable for limited bandwidth, while adaptive methods balance quality and resource use better. Future work will focus on applying machine learning at the edge of the network for predicting user behavior and to pre-process the video frame to find interesting subjects to be privileged during encoding.

Acknowledgment

This work has been partially funded in the framework of the European Union's Horizon 2020 project VITAL-5G co-funded by the EU under grant agreement No. 101016567 and by the EU under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE0000001 - program "RESTART"). The work has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme for the SUNRISE-6G project under Grant Agreement No. 101139257. C. Grasso was supported by European Union (NextGeneration EU), through the MUR-PNRR project SAMOTHRACE (ECS00000022).

References

[1] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6g: A comprehensive survey," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 334–366, 2021.

[2] A. I. Salameh and M. El Tarhuni, "From 5g to 6g—challenges, technologies, and applications," *Future Internet*, vol. 14, no. 4, 2022.

[3] S. Kianpisheh and T. Taleb, "A survey on in-network computing: Programmable data plane and technology specific applications," *IEEE Communications Surveys Tutorials*, vol. 25, no. 1, pp. 701–761, 2023.

[4] C. Grasso, R. Raftopoulos, G. Schembra, and S. Serrano, "H-home: A learning framework of federated fanets to provide edge computing to future delay-constrained iot systems," *Computer Networks*, vol. 219, p. 109449, 2022.

[5] F. Busacca, C. Grasso, S. Palazzo, and G. Schembra, "A smart road side unit in a microeolic box to provide edge computing for vehicular applications," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 1, pp. 194–210, 2023.

[6] H. Yu, M. Shokrnezhad, T. Taleb, R. Li, and J. Song, "Toward 6g-based metaverse: Supporting highly-dynamic deterministic multi-user extended reality services," *IEEE Network*, vol. 37, no. 4, pp. 30–38, 2023.

[7] A. Hazarika and M. Rahmati, "Towards an evolved immersive experience: Exploring 5g- and beyond-enabled ultra-low-latency communications for augmented and virtual reality," *Sensors*, vol. 23, no. 7, 2023.

[8] G. Colajanni, P. Daniele, L. Galluccio, C. Grasso, and G. Schembra, "Service chain placement optimization in 5g fanet-based network edge," *IEEE Communications Magazine*, vol. 60, no. 11, pp. 60–65, 2022.

[9] A. Caruso, C. Grasso, R. Raftopoulos, and G. Schembra, "An adaptive closed-loop encoding vnf for virtual reality applications," in *2024 IEEE 10th International Conference on Network Softwarization (NetSoft)*, (Saint Louis, MO, USA), pp. 222–230, 2024.

[10] "Slices-ri project." https://www.slices-ri.eu/ [Accessed: July 2024].

[11] "Sunrise-6g project," 2024. https://sunrise6g.eu/ [Accessed: July 2024].

[12] M. T. Islam, C. E. Rothenberg, and P. H. Gomes, "Predicting xr services qoe with ml: Insights from in-band encrypted qos features in 360-vr," in *2023 IEEE 9th International Conference on Network Softwarization (NetSoft)*, (Madrid, Spain), pp. 80–88, 2023.

[13] J. V. d. Hooft, M. T. Vega, S. Petrangeli, T. Wauters, and F. D. Turck, "Tile-based adaptive streaming for virtual reality video," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 15, dec 2019.

[14] A. Caruso and G. Schembra, "Impact of user's movements in an immersive 360° video streaming with differentiated compression," in *2024 IEEE 67th International Midwest Symposium on Circuits and Systems (MWSCAS 2024)*, (Spingfield, Massachussez, USA), 2024.

[15] A. Caruso and G. Schembra, "A vr 360°-video encoding framework with differentiated tile compression based on digital-twin technology," in *IEEE International Conference on Multimedia Information Processing and Retrieval (MIPR) 2024*, (San Jose, California, USA), 2024.

[16] K. Trichias, G. Landi, E. Seder, J. Marquez-Barja, R. Frizzell, M. Iordache, and P. Demestichas, "Vital-5g: Innovative network applications (netapps) support over 5g connectivity for the transport & logistics vertical," in *2021 IEEE EuCNC/6G Summit*, (Porto, Portugal), pp. 437–442, IEEE, 2021.

[17] N. Slamnik-Kriještorac, G. Landi, J. Brenes, A. Vulpe, G. Suciu, V. Carlan, K. Trichias, I. Kotinas, E. Municio, and A. Ropodi, "Network applications (netapps) as a 5g booster for transport & logistics (t&l) services: The vital-5g approach," in *2022 IEEE EuCNC/6G Summit*, (Grenoble, France), pp. 279–284, IEEE, 2022.

[18] V. Charpentier, N. Slamnik-Kriještorac, J. Brenes, A. Gavrielides, M. Iordache, G. Tsiouris, L. Xiangyu, and J. M. Marquez-Barja, "Dynamic and Quality-aware Network Slice Management in 5G Testbeds," in *2023 IEEE 279C/6G Summit*, (Gothenburg, Sweden), pp. 611–616, 2023.

[19] Mettlre, "Free 360 video downloads page master series." https://www.mettle.com/360vr-master-series-free-360-downloads-page/. Accessed: February 2023.

[20] R. I. T. da Costa Filho, M. C. Luizelli, M. T. Vega, J. van der Hooft, S. Petrangeli, T. Wauters, F. De Turck, and L. P. Gaspary, "Predicting the performance of virtual reality video streaming in mobile networks," in *Proceedings of the 9th ACM Multimedia Systems Conference*, (Amsterdam, Netherlands), pp. 270–283, 2018.