Conference Paper

Demo Object detection under 5G-edge mob

Marco Araújo
João Silva
Pedro Miguel Santos
Himanshu Singh
Deepak Gunjal,João Fonseca
Paulo Duarte

CISTER-TR-230610

Demo Object detection under 5G-edge mob

Marco Araújo, João Silva, Pedro Miguel Santos, Himanshu Singh, Deepak Gunjal,João Fonseca, Paulo Duarte, Bruno Mendes, Raul Barbosa, Peter Steenkiste, Saeid Sabamoniri, Luis Puente Lam, João Pereira, Harrison Kurunathan

CISTER Research Centre

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail:

https://www.cister-labs.pt

Abstract

In the mid-term future, vehicles will generate large amounts of data for both standalone usage (e.g., to recognize road features and external elements such as lanes, signs, and pedestrians) and cooperative usage (e.g., lane merging). However, processing the captured video and image data results comes with significant computational requirements (e.g., GPUs). Computer vision tasks, such as feature extraction, are unfeasible from a business perspective if performed directly in the User Equipment (UE), as automotive manufacturers are unwilling to increase the end-product 19s costs. Thus, the logical solution is to collect and upload this data to be processed elsewhere. Nonetheless, processing the data as close to the vehicle is important due to latency constraints, thus calling for the use of Mobile Edge Computing (MEC). An additional benefit of this scenario, in which 5G connectivity enables data to be offloaded to the edge, is that the data from our car is not processed alone. Data from several sources, e.g., multiple vehicles and fixed cameras, can be offloaded to the edge node and processed together, enhancing its quality as more sources of data enhance the prediction output of machine-learning models. This demo showcases a video recording from a vehicle uploaded to an edge node via 5G software-defined-radio FPGA devices. There, a YOLO application to detect objects processes the video and communicates this information to the vehicle, ensuring QoS metrics even when the UE performs handover to a different cell or geographical area.

Demo: Object detection under 5G-edge mobility

Marco Araújo*, João Silva*, Pedro M. Santos‡, Himanshu Singh†, Deepak Gunjal†,
João Fonseca*§, Paulo Duarte*, Bruno Mendes*§, Raul Barbosa*§, Peter Steenkiste¶
Saeid Sabamoniri‡, Luis Lam‡, João Pereira‡, Harrison Kurunathan‡
*Capgemini Engineering, Porto, Portugal
†Capgemini Engineering, Gurugram, India
‡CISTER Research Center, ISEP, Porto, Portugal
§University of Aveiro and Instituto de Telecomunicações, Aveiro, Portugal
¶Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA

***Abstract*—In the mid-term future, vehicles will generate large amounts of data for both standalone usage (e.g., to recognize road features and external elements such as lanes, signs, and pedestrians) and cooperative usage (e.g., lane merging). However, processing the captured video and image data results comes with significant computational requirements (e.g., GPUs). Computer vision tasks, such as feature extraction, are unfeasible from a business perspective if performed directly in the User Equipment (UE), as automotive manufacturers are unwilling to increase the end-product's costs. Thus, the logical solution is to collect and upload this data to be processed elsewhere. Nonetheless, processing the data as close to the vehicle is important due to latency constraints, thus calling for the use of Mobile Edge Computing (MEC). An additional benefit of this scenario, in which 5G connectivity enables data to be offloaded to the edge, is that the data from our car is not processed alone. Data from several sources, e.g., multiple vehicles and fixed cameras, can be offloaded to the edge node and processed together, enhancing its quality as more sources of data enhance the prediction output of machine-learning models. This demo showcases a video recording from a vehicle uploaded to an edge node via 5G software-defined-radio FPGA devices. There, a YOLO application to detect objects processes the video and communicates this information to the vehicle, ensuring QoS metrics even when the UE performs handover to a different cell or geographical area.**

***Index Terms*—MEC, 5G, V2X, Object detection, YOLO**

I. INTRODUCTION

The fifth generation (5G) of mobile networks has been the subject of much discussion and research in recent years [1] [2] [3]. Autonomous driving (AD), that involves computationally demanding tasks such as real-time image processing, is a pertinent real-world application that can benefit greatly from 5G and Mobile Edge Computing (MEC). An important use-case with stringent low latency requirements is object detection (OD) for vehicular services, such as emergency braking or traffic management in intersections. Such detection task can be performed in real-time on the vehicle, which can then use the detection results immediately to swerve from road-users with conflicting trajectories, or in the road-side infrastructure, which can then warn nearby road-users about potential collisions in scenarios where road-users cannot see each other (e.g., blind corner). However, when vehicular scenes are too complex (e.g., too many road-users) and the vehicle/road-side infrastructure does not have sufficient processing power, such service can be offloaded to an external edge computing infrastructure. Edge-hosted applications have access to a much wider breadth of resources, but 5G is essential in achieving low communication latency [4], [5].

This demonstration showcases an edge-hosted application using a 5G network. We couple Capgemini's 5G Core (5GC) with the edge to meet the lowest possible latency needs of Vehicle-to-everything (V2X) scenarios using an object detection (OD) application as a proof-of-concept. The OD functionality is provided by the YOLO library [6] applied to video scenes captured in vehicular environments.

II. SYSTEM ARCHITECTURE

This demo uses the Intelligent Edge Application Platform (IEAP) from Capgemini to showcase the potential of 5G applications running on edge nodes. The high-level architecture is illustrated in Fig. 1.

The IEAP is a Mobile Edge Computing (MEC) platform that instantiates and manages edge applications on distributed edge sites or datacenters connected to the 4G/5G mobile network.

The IEAP performs matchmaking of the edge applications' compute resources and end-to-end Quality-of-Service (QoS) communication requirements with the edge nodes, i.e., the platform provides end users with optimum quality of experience. Edge applications can be instantiated via the North Bound Interface (NBI) through the Portal and REST APIs. For example, if there are multiple edge nodes with different compute resources and a computer vision application for connected vehicles requires Graphics Processing Unit (GPU) processing, the IEAP MEC orchestrator will select an edge site from the subset hat can provide the requested acceleration, e.g., GPUs. In addition, it may also verify that the edge site meets the QoS requirements of the application, such as

---

The authors wish to acknowledge and show their public appreciation towards the COMPETE 2020, Portugal 2020 - Operational Program for Competitiveness and Internationalization (POCI), European Union's ERDF (European Regional Development Fund) for providing the necessary co-funding of the projects FLOYD: 5G/SDN Intelligent Systems For LOw latencY V2X communications in cross-Domain mobility applications (POCI/01/0247/FEDER/047222) and OREOS: Orchestration and Resource optimization for rEliable and lOw-latency Services (POCI/01/0247/FEDER/047222).

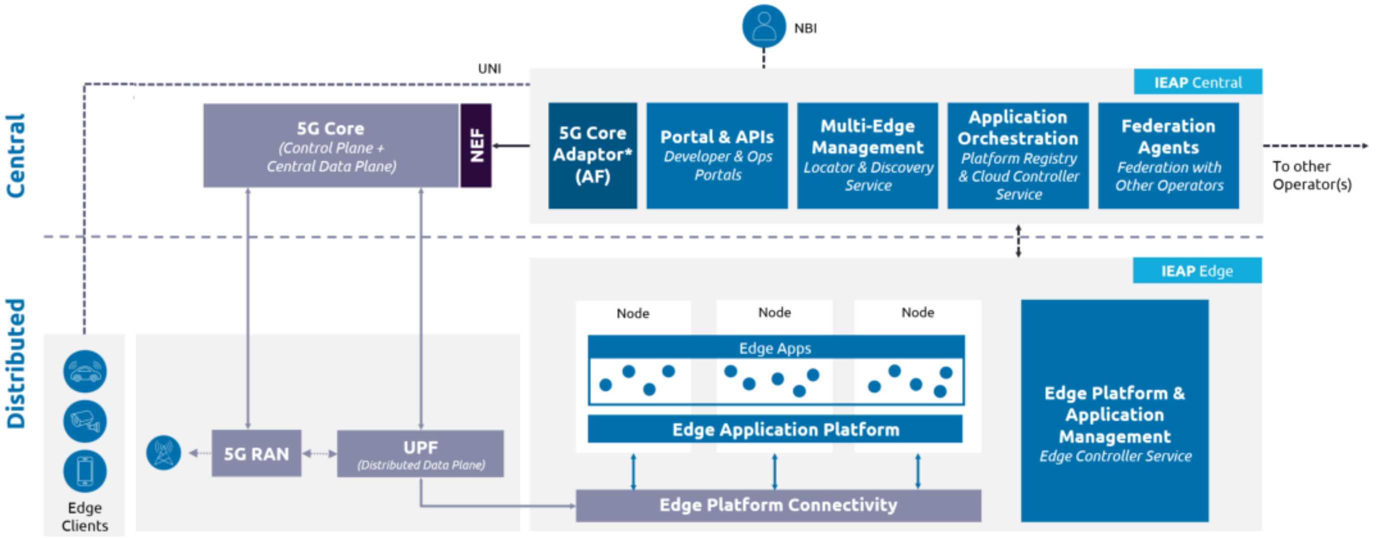

Fig. 1. Capgemini's Intelligent Edge Application Platform

bandwidth, latency, packet loss or jitter thresholds, to serve vehicles connected to a specific cell ID or tracking area.

The user clients or the applications within the client devices communicate with the IEAP Central through the User Network Interface (UNI) standardized in GSMA for various purposes, e.g., discover adequate edge sites for a given application, provide feedback of QoS observed at the device, or request edge application relocation due to device or user mobility.

The IEAP Central contains the *5G Core Adaptor* block which acts as an Application Function (AF) to communicate with the 5G Network Exposure Function (NEF) over the N33 interface to trigger certain scenarios, such as setting the QoS provisioned for the Packet Data Unit (PDU) session carrying edge traffic and/or to retrieve location related events for a moving device connected to 5G network. The NEF is important for MEC applications since it is responsible for communication with the 5G Core network functions for tasks such as the Access and Mobility Function (AMF), Session Management Function (SMF) or Policy Control Function (PCF) to control the communications between the edge and the User Plane Function (UPF). Without the NEF, the AF or *5G Core Adaptor* would have to communicate directly with the individual 5G core network functions. This may involve multiple touchpoints between the platform and the mobile network and integrating the AF with the mobile network would be more complex.

When an operator configures the network, the IP configuration for how the MEC communicates to the UPF through the N6 interface (e.g., via intermediate network routers or gateways) is a priori-defined. For example, if a particular site connected to a specific UPF is responsible for serving a certain traffic area code or cell ID, this will constitute a deciding factor on how routing will happen. If an application client on the user device wants to talk to an application on a particular edge, the MEC orchestrator, over the UNI interface, retrieves information about the user location and the cell ID and uses this information to determine which edge node can serve this particular user by looking at the traffic routing rules and the IP pool of the packet that comes from the user device. When the application responds, it uses the destination IP of the User Equipment (UE) configured into the edge node side, and the packet is sent back to the responsible gateway, which the operator a priori configured in the network.

The IEAP central has a North Bound Interface (NBI) that can be used to connect to a global, higher-level, Management and Orchestration (MANO) system such as ONAP, which at the MEC level can, for example, be responsible for creating edge nodes on available infrastructure (which then would become visible to the MEC via the Multi-Edge Management block of the IEAP) and collecting the underlying infrastructure metrics.

The UPF and other network functions of the 5GC are bundled in Capgemini's Virtual Next-Generation Core (ViNGC) software, which is an optimized virtualized core network for 5G stand-alone deployments. Its core network functions can run as containerized applications within public or private clouds, and in cloud-native environments. The core is built on a Service-Based Architecture (SBA), where the network components announce themselves and offer services that can be used by other network elements in the core through APIs.

The Radio Access Network (RAN) and mobile device clients can use an open-source 3GPP RAN protocol stack, such as OpenAirInterface (OAI) RAN [7] or srsRAN. OAI and srsRAN provide a flexible and customizable RAN solution that can be used for research, experimentation, and commercial deployment of 5G networks. It contains several components, including the Physical Layer (PHY), the Medium Access Control Layer (MAC), the Radio Resource Control (RRC) layer, and the Packet Data Convergence Protocol (PDCP) layer.

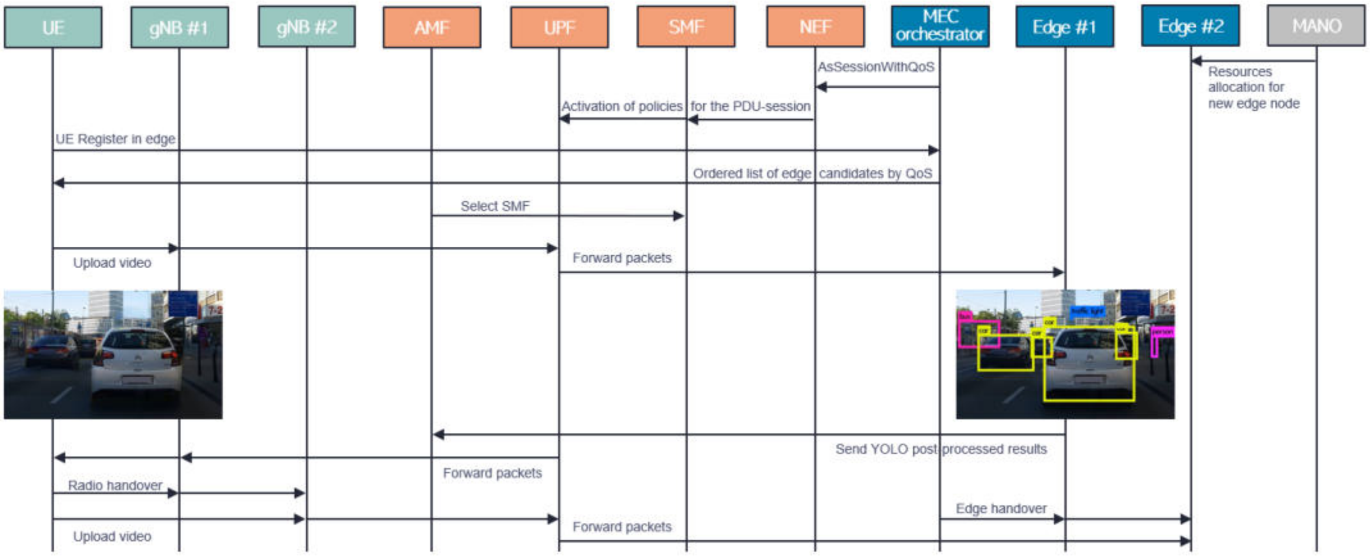

Fig. 2. Call-flow diagram

III. DEMONSTRATION

The demo shows an object detection application running in an edge node. ONAP, acting as the MANO system, uses available compute resources to request the creation of a new edge node. IEAP central, in its role of MEC orchestrator, instantiates the YOLO application on this new edge node. Then, via the N33 interface towards the NEF, it uses AsSessionWithQoS, a RESTful API that defines a set of data models, resources and the related procedures, to create and manage the PDU data path session with the required QoS connecting the UE with the YOLO application running in the edge.

When a vehicle, in its role as the UE using the YOLO application as a client, registers in the network, it retrieves the list of edge nodes via the UNI interface, with the goal of discovering and selecting an edge node that can meet its requirements. It then triggers the AMF in the 5GC to select the SMF and UPF network functions that work best with the YOLO application that is running in the edge server node, given the required QoS metrics.

In the event of a radio handover, the MEC orchestrator observes that the QoS requirements can no longer be met and it triggers an edge handover to an edge closer to the UE, thus continuing to allow to tune the QoS of the video streams being received by the network. Fig. 2 presents the call-flow diagram in this demo.

Both the UE and the gNB are implemented on a Software Defined Radio (SDR), more precisely an Ettus USRP B210 for the UE and a National Instruments USRP 2954R for the gNB, running OAI [8] or srsRAN. The 5GC and the MEC are, respectively, ViNGC and IEAP, both from Capgemini.

The OD service, provided by the YOLO library, is applied to a video of a selected vehicular scenario, such as imminent collision (requiring emergency braking) or equivalent. The captured scene can be either from the point-of-view (PoV) of the front-facing camera of a vehicle in motion or the PoV of a road-side camera monitoring an area of interest. We resort to a scene captured in a controlled environment using a testbed of scaled-down autonomous vehiclesAn action may be showcased in reaction to detections.

IV. CONCLUSIONS

Our work demonstrates a V2X use case integrating MEC and ETSI NFV in 5G networks to perform data offloading with reduced end-to-end latency and QoS assurance under mobility and handover scenarios. While the application in this demo is image processing, our work is applicable to any dedicated V2X MEC service within a 5G framework.

REFERENCES

[1] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, “A survey on low latency towards 5g: Ran, core network and caching solutions,” *IEEE Comm’s Surveys & Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018.

[2] N. Hassan, K.-L. A. Yau, and C. Wu, “Edge computing in 5g: A review,” *IEEE Access*, vol. 7, pp. 127276–127289, 2019.

[3] M. A. Habibi, M. Nasimi, B. Han, and H. D. Schotten, “A comprehensive survey of ran architectures toward 5g mobile communication system,” *IEEE Access*, vol. 7, pp. 70371–70421, 2019.

[4] J. Soriano, G. Jiménez, E. Correa, and N. Ruiz, “Challenges in the automotive software supply chain, connected car : Benefits from an intent policy framework,” in *2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR)*, pp. 1–5, 2021.

[5] D. Sabella, P. Thakkar, L. G. Baltar, J. Sharma, and et. al, “Global mec supporting automotive services: from multi -operator live trials to standardization,” in *2021 IEEE Conference on Standards for Communications and Networking (CSCN)*, pp. 7–13, 2021.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[7] F. Kaltenberger, A. P. Silva, A. Gosain, L. Wang, and T.-T. Nguyen, “Openairinterface: Democratizing innovation in the 5g era,” *Computer Networks*, vol. 176, no. 107284, 2020.

[8] F. John, J. Schuljak, L. B. Vosteen, B. Sievers, A. Hanemann, and H. Hellbrück, “A reference deployment of a minimal open-source private industry and campus 5g standalone (sa) system,” in *IEEE 10th Int’l Conf. on Inform., Comm. and Networks (ICICN)*, pp. 1–9, 2022.