Exploring the suitability of a digital twin- and extended reality-based telepresence platform for a collaborative robotics training scenario over next-generation mobile networks

Exploring the suitability of a Digital twin- and eXtended Reality-based telepresence platform for a collaborative robotics training scenario over next-generation mobile networks

Davide Calandra, Filippo Gabriele Pratticò, Jacopo Fiorenza, Fabrizio Lamberti
Dipartimento di Automatica e Informatica, Politecnico di Torino, Turin, Italy
{davide.calandra, filippogabriele.prattico, jacopo.fiorenza, fabrizio.lamberti}@polito.it

***Abstract*—This paper explores the application of Digital Twins (DTs) and eXtended Reality (XR) in the context of Industry 4.0, and investigates new ways in which these technologies can be used in remote assistance/training scenarios involving Collaborative Robots (CRs). The study builds upon a previous work that examined the suitability of a novel DT/XR-based telepresence platform for CR programming in terms of network capabilities. The present work addresses some of the limitations of the previous study in the context of a new use case, integrating human pose estimation and object tracking to enhance the DT functionalities of the platform, and testing it in a riveting task scenario. The results in terms of bandwidth and latencies obtained in a laboratory setup emulating 6G performance show the potential of DTs and XR in supporting the collaboration of distant human operators and CRs over future mobile networks, paving the way for the development of new services for next-generation industry.**

***Index Terms*—Digital Twin, Telepresence, Collaborative Robots, Industry 4.0, Augmented Reality, Virtual Reality, Human Pose Estimation, Training**

I. Introduction

In the context of Industry 4.0 – a term that indicates the rapid change in technology and industry due to the introduction of smart processes and interconnected devices – a prominent role has been played by Collaborative Robots (CRs) or *cobots*. In fact, being designed to work closely with humans, these devices made novel production processes based on new forms of Human-Machine Interaction (HMI) possible [1]. In these scenarios, other technologies like, e.g., Internet-of-Things (IoT) and eXtended Reality (XR) are typically involved for visualization and control purposes [2].

The key advantages of XR technology, which encompasses Virtual Reality (VR), Augmented Reality (AR), and Mixed Reality (MR), are not related only to visualization and HMI, but also to the possibility to create shared collaborative environments between distant users.

Focusing on the specific field of industrial manufacturing and considering immersive applications of these concepts, some aspects are still the subject of intense investigation from both industry and academia. Firstly, many times it is hard to completely reconstruct both the local user and the physical environment without employing complex or unwieldy setups. Secondly, the limitations of current networks in terms of data rates and latencies could bring to a poor system performance and, hence, to a bad User eXperience (UX). The latter issue will become more and more critical with the advent of new smart appliances for industrial IoT, e.g., capable of providing their Digital Twin (DT). In this regard, the development of next-generation mobile networks will play a fundamental role in supporting these increasingly demanding DT-based scenarios, also paving the way to the adoption, in this context, of important technology enablers like Artificial Intelligence (AI) and Machine Learning (ML) [3].

Given these premises, the present work builds upon a previous study that analyzed the suitability of a novel DT- and XR-based telepresence platform for CR programming from the point of view of network dependability aspects [1]. Therein, platform requirements in terms of latency and bandwidth were analyzed by comparing the performance that could be achieved on current and next-generation mobile networks. The results of that study, obtained by deploying the devised platform on a laboratory setup emulating the peak performance of a future 6G network, showed the partial impossibility to use currently available mobile networks (4G and 5G), and the actual feasibility on the forthcoming ones (6G).

Moving from the findings of the previous investigation, the present paper operates an extension of the original analysis by tackling some of its drawbacks. In particular, the reference work did not leverage some of the previously mentioned technology enablers (AI/ML), whose usage in the considered scenarios is expected to grow thanks to the capabilities of next-generation networks. Furthermore, the reference use case was only partially representative, since it completely revolved around an approach for programming CRs that was specifically conceptualized for the purpose of the investigation (to stress network aspects) but does not belong to common practice.

The present work defines a new and more relevant use case in the industry field, i.e., a remote assistance/training scenario [4], [5]. In the context of the Industry 4.0 transition, it can be

This work has been partly funded by the European Commission through the H2020 Hexa-X project (Grant Agreement no. 101015956).

assumed that, for a wide range of tasks previously performed by humans, the CRs will find ample room of employment as a support tool for human operators. Thus, in the new use case, a Local User (LU) – sharing the workspace with a CR – is an operator which is highly experienced in the context of a given task but has no experience regarding how to perform that task in collaboration with the CR. A second, distant user – the Remote User (RU) – plays the role of the expert, who is in charge of teaching the other peer whatever is necessary to perform the task by collaborating with the CR (in terms of procedural steps and safety notions).

In order to properly support the new scenario, two AI/ML-based functionalities were integrated in the XR platform. Fistly, a full-body estimation of the LU was used to provide the RU with a more precise DT of the LU. To this aim, a multi-camera setup along with Computer Vision (CV) and AI techniques were employed to perform Human Pose Estimation (HPE) on the LU. Secondly, compared to the reference work, a more precise DT of the workpiece in terms of localization (position and rotation) and visualization was obtained through a 6-Degrees-Of-Freedom (6-DOF) ML-based tracking approach. Finally, the platform was tested in a use case where the CR has to perform a riveting task on a metal plate and the LU has to hold the workpiece; the RU, as said earlier, supervises the work of the LU.

II. Related Works

In the field of collaborative robotics, the technological progress has enabled machines to perform novel tasks and engage in new forms of HMI. As the usage of CRs in industrial operations is growing steadily, industry and academia evaluate constantly new systems that can improve human-robot collaboration [6], [7]. At the same time, the combined use of CRs and XR has been extensively investigated too.

Pérez et al. [8], for instance, proposed a VR platform to train industrial robot operators. The objective was to demonstrate the effectiveness of a virtual training system encompassing an Head-Mounted Display (HMD) and a DT of the robot in terms of costs, safety and optimization. The comparison was made against a traditional training approach, with a real KUKA KR500-2 robot controlled through a SmartPad. Besides showing the advantages of VR training in the above dimensions, the devised approach additionally allowed to reduce the amount of time the robot was occupied for training purposes, increasing the industrial production. The main disadvantage was represented by the need to create the DT of the real environment for each different use case.

When it comes to industry and manufacturing, the interaction between an operator and the CR often involves other elements, such as tools and workpieces, or components in assembly tasks. As a matter of example, the work by Newbury et al. [9] integrated the tracking of a generic object during a handover task. The main goal was to demonstrate the helpfulness of communicating intentions of the CR to the user in advance. This goal was reached by developing a system including a robotic manipulator, a Microsoft HoloLens 2 MR device, and a RGB-D camera mounted on the end-effector of the robot. Thanks to ArUco markers placed on each side of the object to be tracked (a cube, in this case), the user could visualize a wireframe representation of it in the MR interface. Moreover, the system allowed the user to perform the handover of the tracked object to the robotic harm by visualizing a preview of the gripper final position.

Indeed, approaches in the context of collaborative robotics like those described above, involving distant users that can be either located within the CR workspace or be off-site (remote) and combining different XR technologies, may come with particularly demanding requirements in terms of data rates and end-to-end latencies. To cope with the possible limitations set by current network technologies, some works started to consider the performance of future 6G networks [10], [11].

In particular, in [1], Calandra et al. performed an analysis that represents a foundation for the present paper. In that work, which was carried out in the context of the European flagship project named Hexa-X [12], a prototype implementation of a DT- and XR-based telepresence platform was evaluated in terms of network dependability requirements. In the selected evaluation scenario that concerned collaborative CR programming, a LU could interact with a MR interface visualized through a HMD (an HoloLens 1) to directly program a CR (a KUKA LBR Iiwa 14 R820) in a fictional pick & place scenario. At the same time, a RU could join the session by using a VR HMD (a Meta Quest 2) capable to visualize the DTs of both the CR and the LU, along with the reconstruction, in the form of a point cloud, of the working area. The LU starts the programming of the CR. When needed, the LU can pass the control to the RU, along with the ability to create "slices" of the program on a virtual replica of the CR to be integrated (once previewed and accepted by the LU) in the main program. The two peers can follow this "alternate programming" approach till the completion of the task.

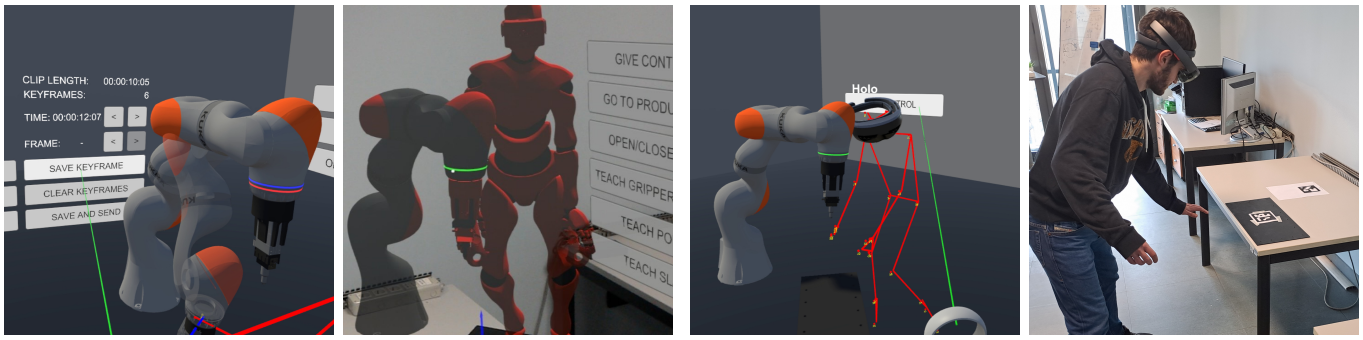
This scenario was evaluated by deploying the platform on a laboratory setup emulating the peak performance of 6G by means of currently available Wi-Fi and wired network technologies. The results of the analysis showed the partial impossibility to use the platform on current-generation mobile networks, due to their limitations in terms of latency (for 4G) and bandwidth (for both 4G and 5G). The theoretical performance of 6G [13], however, was found to be enough for guaranteeing feasibility on next-generation networks [1].

III. Materials and Methods

As mentioned, the methodology pursued in this work was inspired to that presented in [1]. The XR platform detailed in the reference work was used as a basis for the new investigation, and extended to introduce the AI/ML-based techniques supporting the new, training-oriented use case.

A. Remote Training Use Case

The selected scenario deviated from a remote collaboration one, in which both the LU and the RU share the same expertise level, to a remote assistance one, where the LU has to practice



(a) (b) (c) (d)

Fig. 1. Creation of a program clip by the RU to train the LU (a). Avatar of the RU as seen by the LU (b). Full-body DT of the LU represented as a set of connected MediaPipe Pose landmarks (c). LU performing the task with the tracked workpiece (d).

in order to work with the CR for performing a given task under the guide of a remote expert (the RU).

CV and AI based techniques can offer great opportunities for advanced interaction scenarios. Using these technologies to enable new capabilities requires additional hardware (e.g., sensors and processing nodes), which means that network requirements may change. In the original use case, the new DT functionalities could have the effect of improving the visual representation of the local workspace, but would not be mandatorily needed. For this reason, it was decided to shift the focus of the RU away from CR programming and towards the actions carried out by its counterpart. Thus, differently than in [1], the task to be performed does not pertain CR programming anymore.

Another difference with the original setup is the lack of the physical CR. This choice was made since it allows the LU, inexperienced in working with the CR, to practice in a safe environment. [14] To this purpose, the physical robot was replaced with a virtual replica, placed relatively to an ArUco marker; the virtual CR is visible to both the users. In order to properly replicate the behavior of the real CR, as well as to provide the same functionalities of the Sunrise program used in the previous iteration, a porting of the KUKA Sunrise CR client was performed (from Java to C#). Thanks to this CR emulation, the real CR was replaced, with a comparable effect in terms of system performance.

The chosen task that the LU has to perform with the CR is a collaborative operation that consists in the riveting of a metal plate (sized $25\times50\times3$ mm with 18 sockets). The task was selected among those typically relevant for human-robot collaboration [15] and training simulations [16]. In this case, the operator should take a position and keep the plate in place, in order to allow the CR to perform the riveting; the virtual CR was equipped with a riveting flange to pursue this objective.

The workflow of the considered task is as follows:

- The LU informs the RU about the task and the role of the CR in a real production cycle;
- The RU proceeds to program the CR in order to simulate its real behaviour when deployed in the cell;
- The LU can visualize a preview, in form of MR content, and evaluate the CR motion;
- The LU tries to execute the task in collaboration with the simulated CR, while the RU observes and supervises the operations; this is possible through the visualization of the virtual CR, the DT of the workspace and the LU's body pose;
- The LU keeps performing the task until he or she feel it is necessary, or is requested by the RU.

For what it concerns the effect on performance that this task change could bring to the system, it can be hypothesized that the difference in skills between the two users may result in a greater use of the voice chat provided by the platform (with a consequent increase in bandwidth usage related to VOIP), and in a smaller number of programming token passages.

B. Interaction Protocol

The previous platform features that enabled the so-called alternate programming concept (e.g., the possibility to create slices of the program and send them to the other user, to preview the programmed motion, etc.) [1] were taken as a basis for the realization of the new use case; in fact, the original implementation provided a sufficient level of functionality to support a generic remote assistance scenario that could be useful also in a training use case. On the one hand, the RU can take advantage of these features to draft a rough version of the robot program, which will then be executed by the LU (Fig. 1a and Fig. 1b). On the other hand, the LU can perform the assigned task, while the RU can observe the operations and intervene at any time (Fig. 1c and Fig. 1d).

C. Architecture

As said, the original implementation of the XR platform [1] required some modifications in order to support the use case related to the new investigation. Like in the reference work, the goal was to emulate a 6G-enabled mobile network by means of existing wireless (5Ghz) and wired (100Mbps to 10Gbps) networking technologies. The architecture of the modified laboratory setup is reported in Fig. 2.

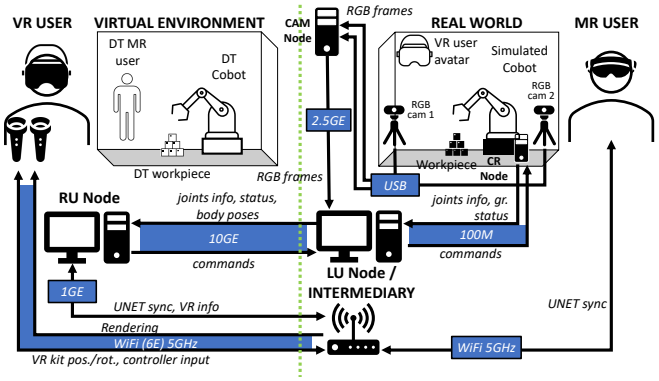


Fig. 2. Architecture deployed in a laboratory setup for the current investigation (modified with respect to [1]).

For the development of the XR platform, it was decided to maintain Unity 2018.4 with the integrated high-level network API (UNET HLAPI). This API was used to implement the network behaviours of each entity following a client-server approach. To deliver the network messages, the asynchronous messaging library ZeroMQ (ZMQ) was used, like in the previous implementation. The LU and RU nodes were again hosted on two PCs with Intel i7-8700 processors and NVIDIA GTX 1080 and 1080 Ti graphics cards, respectively. The LU node also had a 100M Ethernet card for connecting to a node emulating the real robot controller. The *Intermediary* broker application was deployed on the *LU node*, and the *RU node* acted as a ZMQ node and UNET host. The MR and VR HMDs (HoloLens and Quest 2) were connected to the RU node through a 5Ghz Wi-Fi 6E router.

To allow the communication between the XR platform clients and the other components of the system, NetMQ v4.0.1.6 (native C# port of ZMQ) was used to open various channels between the UNET server and the Intermediary, being the UNET architecture server-authoritative.

The MR and VR experiences were not significantly modified in terms of interface and UX. The original setup, however, leveraged a RGB-D camera, which was removed in the new configuration as it was deemed to lead to an overly cluttered visualization and suffer too much from occlusions when the user got close to the robot. These issues could not be fully solved by adding more cameras of the same kind. Thus, two RGB cameras were used in place of the Intel RealSense to estimate and visualize the pose of the LU. The cameras were connected to a dedicated node (*CAM node*), linked to the Intermediary with a 2.5GE Ethernet dongle.

D. Workpiece Tracking and HPE

In the reference work, the point cloud of the workspace indirectly provided a reconstruction of the workpiece too. However, the result of using an RGB-D sensor was not clear and precise enough due to the noisy nature of the point cloud. Hence, for the implementation supporting the present study it was decided to take advantage of the RGB video feed from the MR HMD, which the LU constantly wears.

For what it concerns the tracking algorithm, a marker-based solution (ArUco [17]) was adopted, as it was already employed to perform the initial alignment between the real-world and the holograms. The Mixed Reality ToolKit's (MRTK) Spectator View ArUco detection plug-in was modified to support both the stationary and the moving detection strategies [18]: the first marker was used to align the MR content with the world, whereas the second marker was used to detect the workpiece.

For what it concerns the HPE module, it was implemented after an evaluation of multiple libraries and solutions. Different methods were applied by numerous works to obtain 2D and 3D skeletons to be used in a certain range of applications [19]. Some relevant examples are OpenPose [20], BlazePose/MediaPipe Pose [21] and YOLOv7 [22], each one characterized by pros and cons: YOLOv7 and OpenPose can run multi-person HPE, but they mostly work well if a GPU is available; in turn, MediaPipe runs faster on CPU-based applications and is also able to estimate a 3D pose from a 2D image, but GPU inference is not always supported.

For its lower complexity and higher performance compared to the other two mentioned options, MediaPipe was chosen. Taking advantage of the PyZMQ (Python implementation of ZMQ library) and of the MediaPipe Pyhton solution, a Python client was designed in order to receive a stream from two or more *Camera clients*, generate a 3D pose, and send it to the XR platform through the Intermediary. The *HPE client* was executed on the same machine acting as the Intermediary/LU node, taking advantage of its computational power.

Each of the 32 landmarks provided by MediaPipe comes with a visibility value (0 for not visible, 1 for visible). In order to fuse the 3D poses coming from the two cameras, a rigid transformation (computed on the basis of the four landmarks related to the hips and the shoulders) was used to align the two sets of 3D landmarks. Then, a weighted average (based on the visibility value) for each of the three components (x, y, z) was computed to obtain the resulting skeleton. It is worth observing that each of the components had to be considered as a local coordinate in relation to the body hip, since MediaPipe does not provide information on the actual position of the human body in the 3D world. However, it was possible to get the information about the orientation and position of the LU thanks to the 6-DOF tracking of the MR HMD. In this way, the 3D pose could be moved and rotated to obtain the alignment with the LU's head. The so-calculated 3D pose is then sent through the Intermediary to the XR server running on the RU node, which uses received data to display a representation of the full-body pose in terms of landmarks and connections between them (Fig. 1c and Fig. 1d).

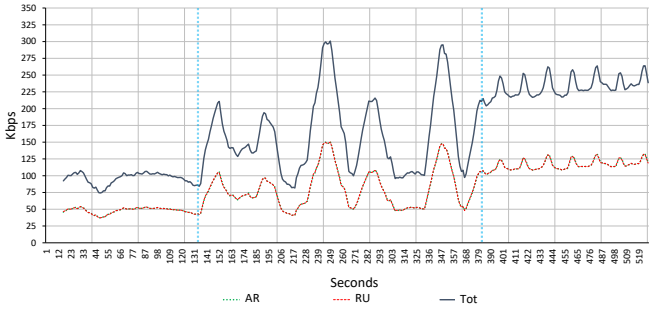
IV. Experimental Results

In this section, the results obtained by evaluating the new platform configuration are reported. The experiment involved a fictional remote training task, where one user played the part of an expert (the RU, in VR) and the other played the part of a local operator (the LU, in MR) inexperienced in terms of human-robot collaboration. As mentioned when

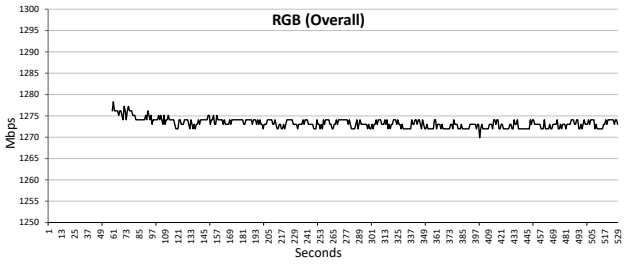
introducing the use case, the riveting of the metal plate was performed by the simulated CR, while the operator had to change his or her position to minimize the risk of collisions. The users were asked to repeat the task 10 times in order to evaluate the platform under possibly different operating conditions (∼10min per execution).

Statistics regarding bandwidth and latency between the network nodes were gathered using custom probe logic included in the developed software for each execution.

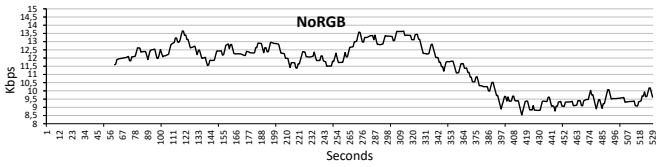
The results concerning the bandwidth occupation during one execution of the task are reported in Fig. 3, whereas the overall results regarding latency can be seen in Fig. 4.



(a) Outbound node bandwidth for the UNET network layer.



(b) Message bandwidth for the ZMQ network layer considering only the RGB data stream.



(c) Message bandwidth for the entire ZMQ network layer traffic excluding the RGB data stream.

Fig. 3. Networking results regarding bandwidth allocation (plots for one execution created using a moving average window of 60s).

For what it concerns the UNET layer, results showed that the latency between the rendering of virtual content on the MR HMD and the same representation in VR was $L_{MR_{HMD}-RU} = 16.94$ms on average, with a minimum of 3.50ms and a maximum of 79.50ms; the average value was lower than that measured in the previous study, whereas the maximum was higher [1] than it, probably due to a

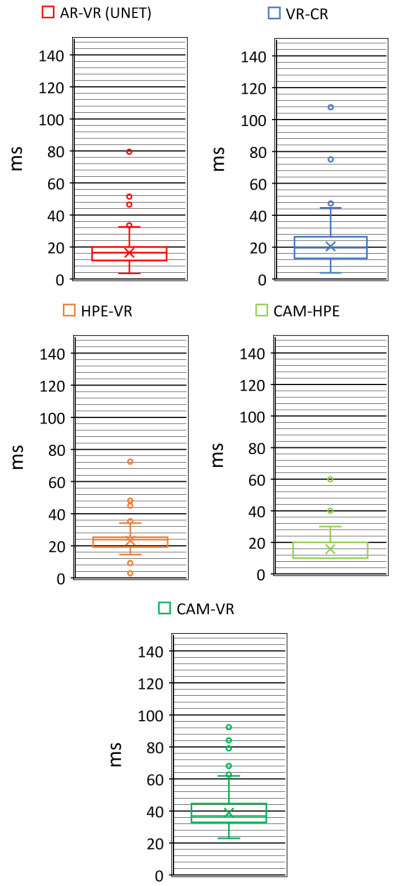


Fig. 4. Networking results regarding latency (all trials).

temporary congestion of the network. It is worth noticing that $L_{MR_{HMD}-RU}$ already includes the render loop latency of the VR application running on the VR node (∼11.1ms), which should be subtracted from the total value, bringing to a $L_{MR_{HMD}-RU} = 5.84$ms. Regarding the bandwidth occupation $B_{MR_{HMD}-RU} = 165.87$Kbps (min 15.81Kbps, max 363.47Kbps), it surpassed the measured value of the previous study, probably because the new scenario requires a higher number of UNET messages to synchronize the LU's representation in terms of landmarks and the workpiece (managed as Unity transforms), along with a larger usage of the VOIP due to the assistance. Moving to analysing the latency between the simulation of the CR and the MR HMD, values were sensibly better than those previously measured, despite the effort in trying to accurately simulate the real CR controller ($L_{CR-MR_{HMD}} = 20.41$ms, min 3.90ms, max 107.70ms). This outcome supports the hypothesis (formulated in [1]) of a malfunction of the network layer of the real CR.

Considering the CAM and HPE clients (respectively running on the CAM and LU nodes), the measured latencies were the one between the two, ($L_{CAM-HPE} = 15.79$ms, min 10.00ms, max 60.00ms), and the one between the HPE and the VR client ($L_{HPE-VR_{HMD}} = 23.24$ms, min 2.90ms, max 72.40ms).

Hence, the total latency between the real movements of the LU and the relative representation in the VR scene in form of "stick figure" was $L_{CAM-VR} = 38.90$ms (min 22.90ms, max 92.40ms). Regarding bandwidth, the results were consistent with the expectations ($B_{CAM-HPE} = 1.27$Gbps, min 1.13Gbps, max 1.36Gbps), considering that the size of 1 second of a stream of two 1280×720 RGB cameras at 30fps is 1.23Gb. Conversely, the stream of processed body poses at 30Hz was not particularly onerous ($B_{HPE-VR} = 0.18$Kbps, min 0.05Kbps, max 0.20Kbps). The same can be said for the ZMQ-related traffic not related to the RGB streams ($B_{NORGB} = 12.68$Kbps, min 0.12Kbps, max 38.30Kbps).

V. Discussion and Conclusions

This paper performed an evaluation of a DT- and XR-based telepresence platform for collaborative robotics built upon the design presented in [1], when applied to a different and more representative industrial scenario, i.e., remote assistance/training. Evaluation focused on network requirements, considering in particular deployability on future 6G networks.

Experimental results showed that, similarly to what found in the previous investigation, the current 4G and 5G networks cannot satisfy the minimum requirements to support the selected scenario, in particular in terms of bandwidth, which surpasses the peak values for real-world implementations of these networks [23]. Regarding latency, current 5G networks provide sufficient performance, but the same does not hold for 4G ones [24]. These limitations will be possibly solved with the advent of next-generation networks, theoretically characterized by significantly better performance in terms of latency and bandwidth (up to 1ms and 1Tbps [13]).

Future works will aim to extend the current analysis to consider UX aspects of human-robot collaboration, by running, e.g., a user study on usability factors. Different approaches to represent the DT of the LU may be investigated too (e.g., a full-body avatar in place of the connected landmarks), and possibly compared among each other. Finally, other relevant use cases may be identified and included in the evaluation, such as those involving the remote control of the CR.

References

[1] D. Calandra, F. G. Pratticò, A. Cannavò, C. Casetti, and F. Lamberti, "Digital twin- and extended reality-based telepresence for collaborative robot programming in the 6G perspective," *Digital Communications and Networks*, 2022.

[2] A. Sanna, F. Manuri, J. Fiorenza, and F. De Pace, "BARI: An affordable brain-augmented reality interface to support human-robot collaboration in assembly tasks," *Information*, vol. 13, no. 10, 2022.

[3] V. Frascolla, M. Hummert, T. Monsees, D. Wübben, A. Dekorsy, N. Michailow, V. Döricht, C. Niedermeier, J. Kaiser, A. Bröring, M. Villnow, D. Wessel, F. Geiser, M. Wissel, A. Viseras, B. Han, B. Richerzhagen, H. Schotten, D. Calandra, and F. Lamberti, *Lesson Learnt and Future of AI Applied to Manufacturing*. River Publishers, 2022, pp. 207–240.

[4] L. F. de Souza Cardoso, F. C. Martins Queiroz Mariano, and E. R. Zorzal, "A survey of industrial augmented reality," *Comp. & Industrial Eng.*, vol. 139, p. 106159, 2020.

[5] D. Calandra, A. Cannavò, and F. Lamberti, "Improving AR-powered remote assistance: A new approach aimed to foster operator's autonomy and optimize the use of skilled resources," *The Int. J. of Advanced Manufacturing Technology*, vol. 114, no. 9, pp. 3147–3164, 2021.

[6] "Innovative human-robot cooperation in BMW group production," https://www.press.bmwgroup.com/global/article/detail/T0209722EN/innovative-human-robot-cooperation-in-bmw-group-production, accessed: [28-February-2023], 2013.

[7] "Human-robot cooperation at Audi," https://www.springerprofessional.de/en/manufacturing/production---production-technology/human-robot-cooperation-at-audi/14221870, accessed: 28-February-2023], 2017.

[8] L. Pérez, E. Diez, R. Usamentiaga, and D. F. García, "Industrial robot control and operator training using virtual reality interfaces," *Comp. in Industry*, vol. 109, pp. 114–120, 2019.

[9] R. Newbury, A. Cosgun, T. Crowley-Davis, W. P. Chan, T. Drummond, and E. A. Croft, "Visualizing robot intent for object handovers with augmented reality," in *Proc. of 31st IEEE Int. Conf, on Robot and Human Interactive Communication (RO-MAN)*, 2022, pp. 1264–1270.

[10] B. Han and H. D. Schotten, "Multi-sensory HMI to enable digital twins with human-in-loop: A 6G vision of future industry," https://doi.org/10.48550/arXiv.2111.10438, accessed: [28-February-2023], 2021.

[11] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.

[12] G. D'Aria *et al.*, "Expanded 6G vision, use cases and societal values – Including aspects of sustainability, security and spectrum," https://hexa-x.eu/d1-2-expanded-6g-vision-use-cases-and-societal-values, accessed: [24-February-2023], 2021.

[13] "6G: Going Beyond 100 Gbps to 1 Tbps," https://www.keysight.com/it/en/assets/7121-1152/white-papers/6G-Going-Beyond-100-Gbps-to-1-Tbps, accessed: [24-February-2023], 2021.

[14] F. G. Pratticò and F. Lamberti, "Towards the adoption of virtual reality training systems for the self-tuition of industrial robot operators: A case study at kuka," *Comp. in Industry*, vol. 129, p. 103446, 2021.

[15] A. Luxenburger, J. Mohr, T. Spieldenner, D. Merkel, F. Espinosa, T. Schwartz, F. Reinicke, J. Ahlers, and M. Stoyke, "Augmented reality for human-robot cooperation in aircraft assembly," in *Proc. of IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, 2019, pp. 263–2633.

[16] R. Müller, M. Vette, A. Geenen, and T. Masiak, "Improving working conditions in aircraft productions using human-robot-collaboration in a collaborative riveting process," in *AeroTech Congress & Exhibition*, 09 2017, pp. 1–7.

[17] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[18] E. Battegazzorre, D. Calandra, F. Strada, A. Bottino, and F. Lamberti, "Evaluating the suitability of several ar devices and tools for industrial applications," in *Augmented Reality, Virtual Reality, and Computer Graphics*, L. T. De Paolis and P. Bourdot, Eds., 2020, pp. 248–267.

[19] S. Kulkarni, S. Deshmukh, F. Fernandes, A. Patil, and V. Jabade, "Poseanalyser: A survey on human pose estimation," *SN Computer Science*, vol. 4, no. 2, p. 136, Jan 2023.

[20] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2021.

[21] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, "BlazePose: On-device real-time body pose tracking," https://arxiv.org/abs/2006.10204, pp. 1–4, 2020.

[22] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," https://arxiv.org/abs/2207.02696, pp. 1–15, 2022.

[23] "5G Performance - European 5G Observatory," https://5gobservatory.eu/info-deployments/5g-performance/, accessed: [24-February-2023], 2021.

[24] "5G vs. 4G: How will the newest network improve on the last?" https://www.digitaltrends.com/mobile/5g-vs-4g/, accessed: [24-February-2023], 2022.