This article has been accepted for inclusion in a future issue of this magazine. Content is final as presented, with the exception of pagination.

Remote Medical Support in Emergency Scenarios: A WebRTC-Based Solution

Sujit Kumar Sahu, Molka Gharbaoui, Anna Lina Ruscelli, Gabriele Cecchetti, Andrea Sgambelluri, and Piero Castoldi

The authors propose a novel system designed to provide remote medical support using web real-time communication technology.

Abstract

In emergency medical scenarios, the lack of immediate access to expert medical advice can result in delayed treatment and increased mortality rates. Timely and effective communication between on-site emergency operators and remote doctors is crucial for the delivery of most medical treatments and for saving lives. This work proposes a novel system designed to provide remote medical support using web real-time communication (WebRTC) technology. The system enables healthcare professionals to offer immediate assistance to on-site operators through real-time audio and video communication. The presented approach leverages the capabilities of WebRTC to facilitate low-latency and peer-to-peer connections, ensuring that critical medical advice is delivered promptly and efficiently. In this article, the architecture of the system and its implementation are detailed, and a first assessment of its performance is provided. Preliminary evaluations, including the deployment of a working prototype under different experimental configurations, demonstrate the system's potential to enhance emergency response outcomes by bridging the gap between remote medical expertise and on-site emergency care. A scalability study is also performed showing the efficiency of the system in the presence of concurrent users.

Introduction

Emergencies and disasters, whether natural or human-induced, have the capacity to overwhelm healthcare systems and disrupt the delivery of vital medical services. The ensuing chaos, surge in patients, and potential damage to healthcare infrastructure often create immense challenges for healthcare providers and their ability to meet the medical needs of affected populations. Remote healthcare services, which encompass telemedicine and telehealth applications, facilitated by distributed wearable and portable devices in delivering healthcare remotely during crises, have emerged as essential tools to ensure timely and effective healthcare, even in presence of chaos and disruptions thus becoming a key feature in facing the above issues [1]. In addition, several new technologies such as the Web Real-Time Communication (WebRTC) framework have emerged as a robust solution to address the technical implementation challenges related to remote communications. More specifically, WebRTC is a cutting-edge technology standardized by the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF). It leverages an open standard and secure peer-to-peer architecture to enable real-time communication directly within web browsers, thus eliminating the need of complex external plugins or downloads [2].

For this purpose, motivated by the critical need to bring expert and remote medical support in emergency scenarios where time and accuracy are of the essence, in this work we describe the new implementation of a tele-assistance system that provides remote medical support using WebRTC technology. The system is specifically developed for remote healthcare services that respond to the needs of affected populations in diverse disaster scenarios, including natural disasters, pandemics, complex humanitarian emergencies, and first aid interventions provided by the emergency service. In fact, in many such situations, on-site operators may lack the specialized knowledge required to make life-saving decisions, and the delay in receiving expert advice can significantly impact patient situations. In this context, a tele-assistance system leveraging WebRTC technology to support audio and video communications can bridge this gap by enabling real-time and high-quality communication between on-site sanitary operators and remote medical specialists. In particular, differently from the current organization of the emergency service where the medical staff can just interact with the healthcare operators through an audio channel, thanks to the proposed system, the doctor can see the patient as if he was present in the scene of intervention, thus verifying his health status and asking, for example, to do particular movements (e.g., the Cincinnati scale for the diagnosis of the stroke that encompasses the analysis of facial expression, limb movement, speech). This supports the pre-triage phase and ensures that crucial medical advice is delivered promptly and efficiently, which enhances the quality of emergency care and reduces response times. This aspect is particularly important especially for time-dependent pathologies such as stroke, heart stroke and acute respiratory failure where the speed of the correct diagnosis can impact on the survival of the patients. Furthermore, differently from other video conference systems used in medical emergencies, which are based on smartphones or tablets [3][4], in our work, the video camera is positioned on the helmet worn by

ISSN: 0163-6804
Digital Object Identifier: 10.1109/MCOM.001.2400185

The authors are with Scuola Superiore Sant'Anna, Italy.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

the operators, which does not interfere with their operation, thus leaving their hands free and allowing them to smoothly interact with the doctors without impacting their emergency intervention.

Throughout this article, we will explore the design features and implementation aspects of the WebRTC-based tele-assistance system. We will also go through the technical underpinnings of WebRTC, highlighting its core features that make it an ideal choice for such applications while ensuring timely access to care and reducing the burden on healthcare facilities and personnel allowing the doctor to remotely support different on-site interventions without moving from the hospital or the Emergency Service Operations Center. Furthermore, we will discuss the architecture and components of the proposed system, emphasizing its capacity to facilitate real-time audio and video interactions thus delivering a seamless, secure, and user-friendly telemedicine experience for both healthcare providers and patients alike. This article extends the work presented in [5], where a tele-tutoring system based on a wearable helmet with a video camera for the audio-video communication was presented. With respect to [5], here the WebRTC-based application has been improved in terms of stability of the application and video calls in order to provide a high-quality audio-video connection. Furthermore, the support of multiple connections from different remote doctors to a single video call issued from a wearable component is introduced to allow, for instance, the consultation with a team of remote specialists (cardiologist, neurologist, etc.) that could be useful in the case of natural disasters, major accidents or particularly complex and serious patients conditions. Finally, a first assessment of the system performance is presented by testing it under different network conditions (e.g., 4G, WiFi) and by performing a scalability study that reflects its usage in different scenarios (e.g., several specialists discussing a complicated diagnosis case, different doctors monitoring the status of different patients).

The article structure is as follows. The next section presents the state of the art. Then we describe the proposed tele-assistance system and details its main components and features. Following that we describe a proof of concept where a prototype of the system is implemented and evaluated. Finally, we draw conclusions.

Related Work

Remote medical support and monitoring has evolved significantly over the past few decades, particularly with the integration of real-time communication technologies thus facilitating consultations, diagnostics, and treatment without the need for physical presence [4, 6]. Applying those remote healthcare services in disaster and emergency scenarios has also attracted attention and several papers in literature showed their importance in providing high-quality assistance to help damaged and injured patients in the shortest possible time [7]. Among those scenarios, telerehabilitation for post-earthquake rehabilitation represents an important aspect of remote assistance and is proposed in [8]. The authors cite few examples of successful implementations in earthquake-affected areas that rely on the use of videoconferencing, mobile devices, and other digital technologies. They also highlight the need for training healthcare providers and patients to effectively use telecommunication technologies to provide telerehabilitation services. However, no concrete details on the implementation of those services is given. [9] investigates telenursing operational possibilities in disasters. Differently to our work where a proof-of-concept is performed in a realistic experimental environment, authors relied on simulated conditions to investigate their approach. Moreover, the telenursing activity was based on the use of an Internet connection and Skype software to exchange information with the qualified nurses far from the disaster area. Although the authors were able to state that telenursing has the potential to support effective disaster response and accelerate the development of remote healthcare services, the scenario and adopted tools remain limited.

Regarding the implementation aspects, one of the early adoptions of real-time telecommunication in medical support was through proprietary video conferencing systems. Those systems, while effective, often lacked accessibility [10]. The introduction of WebRTC technology was a turning point in this context. WebRTC is an open-source project that provides web applications and websites with real-time communication capabilities via simple application programming interfaces (APIs). This technology supports browser-to-browser applications for voice calling, video chat, and file sharing without the need for internal or external plugins. Several studies have demonstrated the efficacy of WebRTC in telemedicine. For instance, in [11] and [12] e-health systems where patients can consult with doctors staying at home are proposed. The systems rely on the simplest feature of WebRTC, that is, the browser to browser communication while in this article we propose a wearable device equipped with a video camera that sends the WebRTC-based video stream to a dashboard thus making our solution independent from the healthcare operator position. A reality-based system that facilitates and supports emergency response by medical emergency teams is proposed in [13]. Similar to our approach, the system is based on a distributed architecture where emergency personnel wear mixed reality glasses that can transmit audio, video, and data streams bidirectionally to medical specialists stationed in a hospital at any distance. The system however relies on Microsoft Hololens 2™ MR devices, which are proprietary. Instead, our solution implements the software in Raspberry Pi computational unit and can be easily customized according to any use case. In addition, it is seamlessly integrated within the helmets that are worn by the medical staff. Finally, another system that relies on WebRTC and Raspberry Pi devices is proposed in [14] to perform remote interactive patient-professional consultations. Although promising, in addition to the lack of several technical details, the solution was tested only on 5 patients, which does not allow to draw significant conclusions on its robustness and scalability features as we do in this work.

Despite the advancements detailed above, challenges remain in deploying WebRTC for emergency medical support. Issues such as system reliability, scalability, and integration with existing medical information systems still need

Emergencies and disasters, whether natural or human-induced, have the capacity to overwhelm healthcare systems and disrupt the delivery of vital medical services. The ensuing chaos, surge in patients, and potential damage to healthcare infrastructure often create immense challenges for healthcare providers and their ability to meet the medical needs of affected populations.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

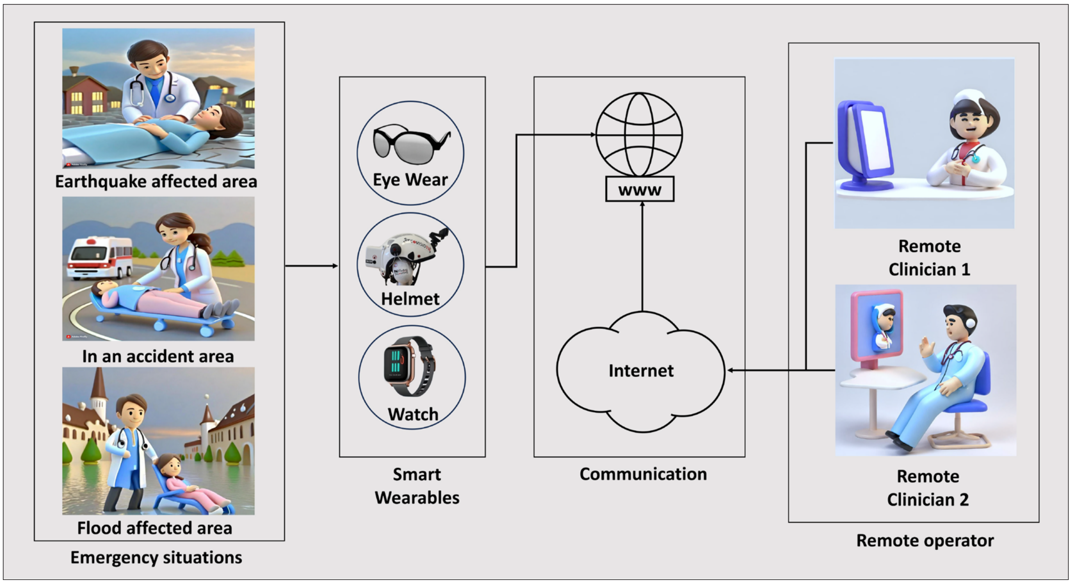

FIGURE 1. Reference Scenario.

to be addressed. The system we propose in this work aims to build on these findings by not only leveraging WebRTC's capabilities but also by implementing a scalable and robust solution to enhance remote healthcare services.

System Description

Remote healthcare services using distributed wearable devices allow for the monitoring of health conditions and delivering of medical care to individuals, regardless of their location. As witnessed during the COVID-19 pandemic, the adoption of telemedicine and wearable devices played a pivotal role in monitoring patients, providing consultations, and reducing the strain on healthcare facilities, while minimizing physical contact. Beyond pandemics, as shown in Fig. 1, various emergency and disaster scenarios, including earthquakes, hurricanes, pandemics, and displaced populations due to conflicts or environmental disasters where traditional healthcare infrastructure can be disrupted and the patients cannot be reached by the doctors, require the adoption of innovative solutions that adapt the use of distributed wearable devices in these distinct contexts, while highlighting their ability to facilitate remote triage, provide early warnings, and offer timely interventions, thereby improving healthcare outcomes and providing remote medical support to on-site operators.

In this context, we propose an advanced evolution of a wearable tele-assistance system that utilizes distributed video cameras as a solution to bridge the gap between the system users (e.g., patients, disaster victims) and the system providers (e.g, clinicians, rescuers), even when physical presence of the medical staff is limited or impractical. More specifically, the system is designed to enhance remote intervention and support the on-site rescuers by allowing remote doctors to see the patients and interact with them.

The tele-assistance system presented in this article mainly relies on the following features:

- Remote Monitoring of patients conditions and medical tele-assistance: The system is equipped with high-definition video cameras that can be worn either by healthcare personnel in remote intervention areas or by also nurses in hospitals (e.g., for remote medical consultation). Those cameras provide a real-time view of the patient's condition and surroundings, enabling healthcare providers to assess the patient's needs effectively and to interact with the on-site rescuers, supporting their actions. The video cameras also allow for remote monitoring of vital signs and physical therapy sessions, enabling healthcare providers to assess progress and make adjustments to treatment plans as needed.
- Two-Way Communication: On-site healthcare personnel and/or rescuers can interact with the remote healthcare providers through the video camera system, enabling real-time conversations, questions, and clarifications about the patient's condition, treatment plan, and recovery process.
- Secure Telecommunication Platform: The system utilizes a secure telecommunication platform to establish a direct audio and video connection between patients and healthcare providers. In this way, it ensures patient data privacy and complies with healthcare regulations.

As shown in Fig. 2, the system is built on three key components, that is, a client, an application, and an online dashboard. The client represents the wearable device, which is the helmet worn by the healthcare operators on-site.

The application acts as the core of the proposed tele-assistance system, which allows the communication to flow seamlessly from the patients to the remote doctors, while securely transmitting data and enabling real-time interactions. It can be installed and configured either on a local Portable Computer (PC) or on a remote one equipped with Internet access. Finally, the online dashboard acts as a bridge for doctors to connect with remote medical personnel and patients through the server. It mainly reproduces a meeting place where healthcare professionals can collaborate effectively, ensuring that patients receive the highest quality care no matter where they are.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

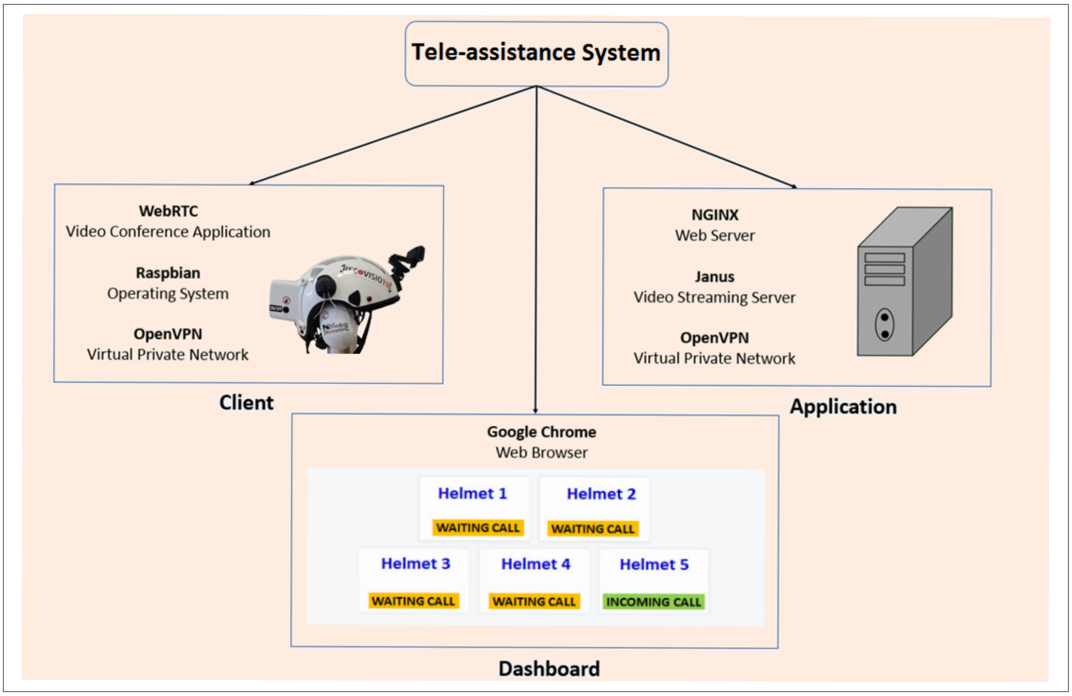

FIGURE 2. Testbed components.

Going deep inside the structure of the system, its main components are described in the following.

Client: The client, shown in Fig. 3, mainly comprises five components. At its core lies the central unit, Raspberry Pi 3b, which is a single board computer widely used for different embedded systems applications ranging from healthcare to smart agriculture and monitoring. With respect to the first version of the system, based on RPI 1, RPI 3b brings a more powerful and faster processor, suitable to meet the computational requirements of the system. In our setup, the RPI handles all computational tasks and runs on the Raspbian operating system which is a Linux-based distribution. It is complemented by a Logitech High Definition camera (C920) for the collection of the visual perspective of the wearer, a headset for the audio interaction, a 0.96-inch OLED (Organic Light-Emitting Diode) display placed on the rear of the helmet to show to the user information about data connection and functioning of the helmet, and a 5000 mAH battery to power up the wearable system. The RPI is connected to Internet for the video call using a Huawei 4G dongle. The operation of the helmet requires a sequence of steps designed for user convenience and efficiency. Indeed, the wearable system was designed to be easily used by the healthcare operators that only need to switch on the helmet, making the connection automatic. The power on of the helmet initiates the boot sequence of the Raspberry Pi which subsequently triggers a background HyperText Markup Language (HTML) code that activates the video conference call feature. This streamlined process eliminates the need for excessive button-clicking by navigating multiple prompts, akin to a conventional video call, thus avoiding distracting the operator from his/her intervention actions. Simultaneously, a Python script runs in the background, displaying the helmet's status on the OLED display, including helmet ID, Internet speed, and Internet connection status. These information are meaningful for the wearer that can deduce if the system can actually support the video call. This is particularly important in remote areas where cellular network coverage can be poor. In this way, our setup showcases the integration of these components, making the illustrated tele-assistance system a user-friendly and accessible solution for both doctors and healthcare providers.

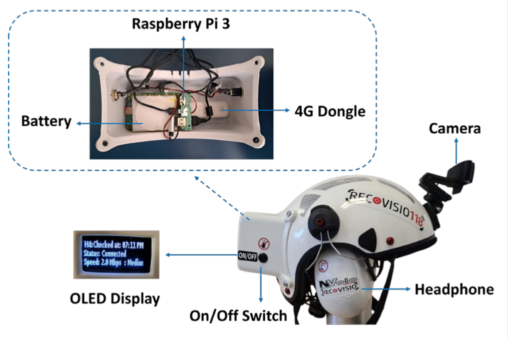

FIGURE 3. Helmet Components.

Dashboard: The dashboard used by the remote doctor is a web browser application, which gives a high-level overview of the tele-assistance system. As shown in Fig. 2, it displays the current status of the helmets (i.e., connected in green/not connected in yellow) to allow the doctors select one of the remote devices and directly get access to the video stream sent by it.

Application: The client (i.e., the helmet) communicates with the application through a Virtual Private Network (VPN) using either the 4G or the WiFi network. The application hosts two separate servers: a web server and a video server. More specifically, any remote specialist provided with the required credentials, can connect to a web browser, select an available client, and then access the content of the video sent by it. In this work, we used the open-source NGINX web

 This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

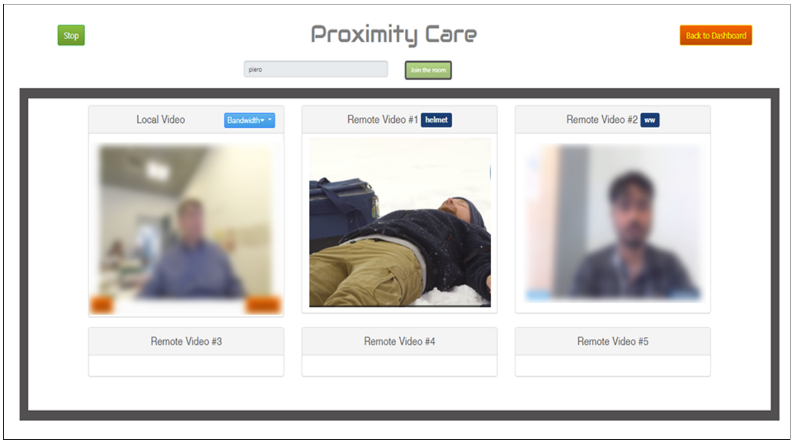

FIGURE 4. Tele-assistance system video room.

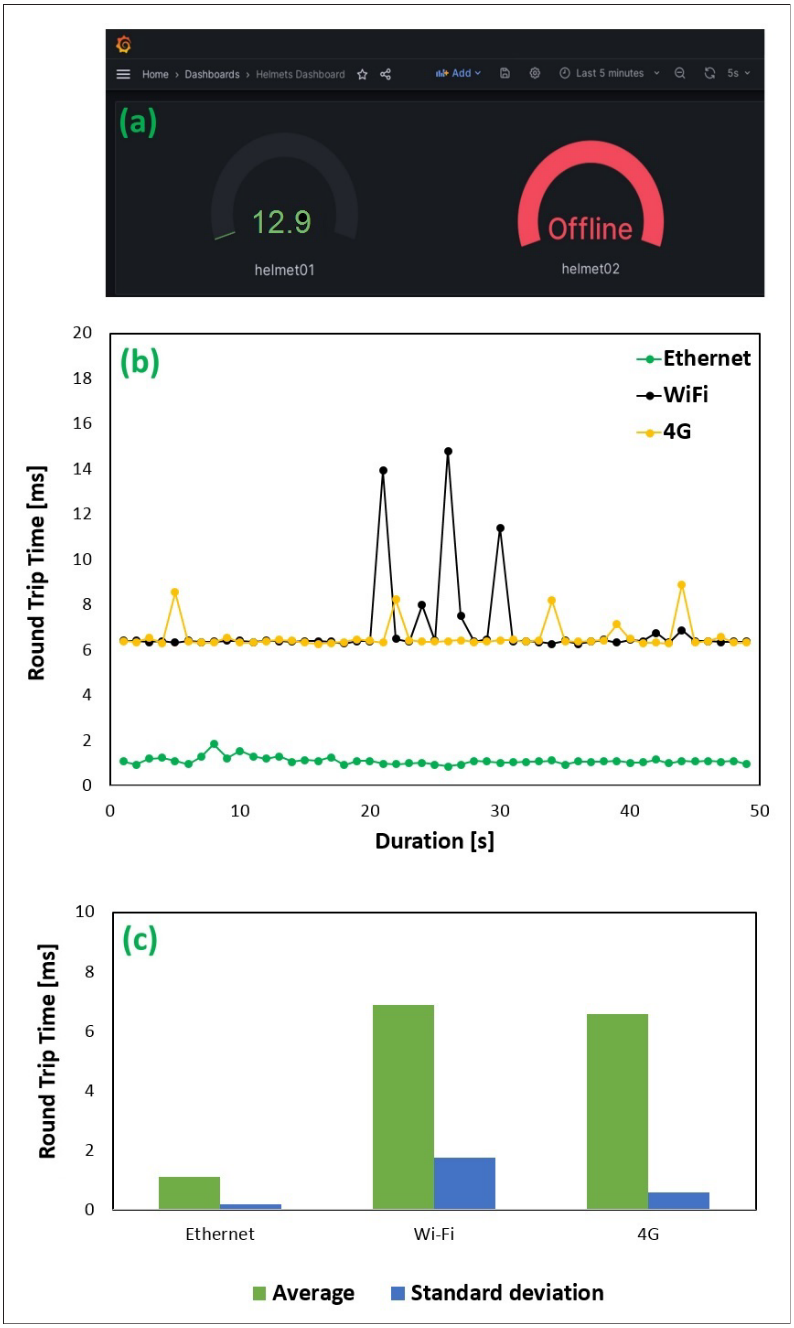

FIGURE 5. a) Grafana Dashboard; b) Round Trip Time; c) Average and Standard deviation of Round Trip Time.

server, which provides Hypertext Transfer Protocol Secure (HTTPS) capabilities and is mainly designed for maximum performance and stability. This allowed us to overcome some of the system's previous version issues where the reconnection to a helmet disconnected because of the lack of 4G coverage was not always successful. Regarding the video server, we relied on Janus, which is an open-source WebRTC server designed and developed by Mettecho [15]. The server simply allows to set up a WebRTC media communication with a browser and sends data to the other side of the communication in real-time. With respect to User space Video for Linux (UV4L), which was adopted in our previous work [5], Janus brings much more stability of the communication, a better video resolution and allows for the concurrent access to the same stream by different doctors.

Once the client powers up, the NGINX server springs into action, transforming the waiting call button on the dashboard into a green-colored button. This is allowed by the openVPN, which creates a secure tunnel linking the NGINX server and the client. When the remote healthcare specialist clicks on the green button, a request is promptly dispatched to the server bank. This request grants access to the remote operators, connecting them seamlessly with the waiting client.

It is worth noting that in this enhanced release of the system the application is designed to support multiple simultaneous calls from different specialists to different clients in remote locations. Moreover, as shown in Fig. 4, several doctors can simultaneously join the same "room" and access the same video stream, which brings further benefits such as the ability for several specialists to remotely discuss difficult cases. This feature is important in the Emergency Service where different simultaneous interventions can require the remote medical support and, for each of them, a tele-consultation with a further remote specialist not present in the Operations Center (for instance a neurologist or a cardiologist).

Proof of Concept

To evaluate the performance and overall quality of the tele-assistance system, we conducted a set of experiments using a set of wearable components. Each experiment has a duration of 5 minutes, which is the average time of a remote medical consultation between a specialist and a patient and/or a nurse as reported by the healthcare operators and doctors during the experimentation of the system within the framework of the Proximity Care Tuscany regional project. In fact, the tele-assistance system is currently in experimentation in the Operations Center of High Tuscany at the Versilia Hospital of Camaiore, where it allows the connection with five healthcare operators teams operating in the remote mountain region of Garfagnana. Upon the start of the experiment, the device is powered on. This process takes 75s on average to complete. Following the startup sequence, the wearable device begins the process of establishing a connection to the server through the OpenVPN protocol. Once this connection is successfully established, the web browser can be accessed to initiate a video call, by simply clicking on the connected device. The video conference promptly starts thereafter.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

For the performance evaluation, we first considered a set of helmets connected to the server, and we evaluated their Round Trip Time (i.e., RTT), which refers to the total time required by a packet to be sent from the helmet, travel over the network to the application and then get back again. Through this metric, we were able to check both the connection availability and latency of our system. To do so, we implemented a containerized application, Telemetry, composed of a polling script and the following open-source tools: InfluxDB and Grafana. The Telemetry application runs on the same machine that hosts our application and periodically executes the Ping command in order to get the RTT. The polling time was fixed to 5s. Collected values were then pushed to the InfluxDB for storage. InfluxDB is an open-source time series database management system for the storage of metrics that are made available for querying by other components of the system. In this work, Grafana is used as the default client of the database to provide a visual and interactive representation of the metric time series. In Fig. 5a we plot the Grafana dashboard with 2 enabled helmets. Helmet01 on the left is connected (Green status) with the last RTT value displayed (i.e., 12.9ms) while helmet02 is shown as "offline" since it was disconnected during the tests.

In Fig. 5b, we plot the RTT collected over 5 minutes for helmet01 when it was connected to three different networks: 4G, Ethernet, and WiFi, while in Fig. 5c, we provide the average and standard deviation values. The obtained results show that the helmet default network (i.e., 4G) provides a stable and low RTT (average value around 6.5ms) thus confirming that we were always able to establish the video communication between the helmet and the application. However, in case the helmet is not able to connect to the 4G network (e.g., poor 4G coverage or not recharged Subscriber Identity Module (SIM) card), it can still connect through any available and previously configured WiFi network. For comparison purposes, we also considered this configuration and directly connected the helmet to a WiFi router, then we measured its RTT. As shown in the results, the obtained RTT values were similar to the ones obtained with 4G although some few spikes that were caused by the instability of the WiFi connection. This was also confirmed by a higher standard deviation value with respect to the 4G configuration. On top of that, we checked with an Ethernet connection, which, as expected, was stable and provided consistently low RTT values.

Next, we performed a scalability study to assess the performance of our tele-assistance system under different conditions. For each test, the video was run for 5 minutes, which, as previously said, represents the average duration of a call during an emergency intervention. The values were collected every 10 seconds and the reported results represent the average of 10 experiment executions. First, we considered a case in which several devices (i.e., PCs, smartphones) are connected through the browser to the same helmet to access the same video stream. This is a realistic scenario where several specialists are called to join the same "room" to discuss a complicated diagnosis case. Table 1 reports the average bitrate of the executed video for each configuration (e.g., 2 concurrent clients, 3 concurrent clients, etc.). Results show that by increasing the

Number of remote doctors simultaneously connected to 1 client (helmet)	**Average bit rate [kb/s]**
1	548
2	566.27
3	577.94
4	561.49
5	572.34
6	582.37
Number of simultaneously connected helmets	**Average bit rate [kb/s]**
1	548
2	552.07
3	532.39

TABLE 1. Scalability study.

number of devices simultaneously connected to the helmet, the average bitrate of the video remains stable. This confirms that the presented system is able to provide stable performance in terms of data rate and, consequently, in terms of Quality of Experience (QoE) of multiple video calls whose quality is not affected by the increasing number of doctors connected to the same video call. These results also demonstrate how the system scales well without any impact on the quality of the video. In a second experiment, we varied the number of helmets simultaneously connected to the dashboard while one doctor was connected to each helmet. This scenario represents the case where several doctors and/or nurses are connected to the dashboard from different workstations and are interacting with different healthcare operators to check the status of different patients. Also in this case results show that the number of simultaneously connected helmets has no impact on the quality of the running videos since the bitrate remains quite stable.

Conclusions

In this article, we presented a tele-assistance system that allows for remote medical support in emergency and disaster scenarios. The system is based on the open-source WebRTC technology, which provides real-time communication between remote clinicians and patients or affected populations. The system was implemented and experimentally tested, and the preliminary results showed its feasibility and efficiency in presence of several concurrent users. In future works some aspects can still be addressed. One example regards the integration of an AI/ML module at the server side that analyzes the video frames in real-time and detects some disease symptoms such as those of stroke (e.g., arm weakness, movements of the arms and facial expressions, speech problems). Another aspect regards hardware with the introduction of more compact components (e.g., smaller battery and Raspberry board) that would fit within the helmet's headphones.

Acknowledgments

This work has been funded by the Cassa di Risparmio di Lucca Foundation through the Proximity Care project and by the Tuscany Region through the TEL-TEAM Project.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

References

[1] V. S. Naresh *et al*., "Internet of Things in Healthcare: Architecture, Applications, Challenges, and Solutions," *Computer Systems Science & Engineering*, vol. 35, no. 6, 2020.
[2] G. Suciu *et al*., "WebRTC Role in Real-Time Communication and Video Conferencing," *Proc. Global Internet of Things Summi*, 2020.
[3] S. Idland *et al*., "From Hearing to Seeing: Medical Dispatchers' Experience With Use of Video Streaming in Medical Emergency Calls — a Qualitative Study," *BMJ Open 12*, vol. 12, 2022, arXiv:https://bmjopen.bmj. com/content/12/12/e063395.full.pdf.
[4] S. Sahu *et al.*, "A Perspective of Telemedicine Videostreaming Systems for Emergency Care," *Proc. 2nd IEEE Int'l. PerCom Workshop on Telemedicine and e-Health Evolution in the New Era of Social Distancing*, 2023.
[5] A. Ruscelli *et al*., "A Medical Tele-Tutoring System for the Emergency Service," *Proc. IEEE Int'l. Conf. Pervasive Computing and Commun. Workshops and other Affiliated Events*, 2021, pp. 410–12.
[6] M. Alshamrani, "IoT and Artificial Intelligence Implementations for Remote Healthcare Monitoring Systems: A Survey," *J. King Saud University — Computer and Information Sciences*, vol. 34, no. 8, Part A, 2022.
[7] N. Mohammadzadeh *et al*., "Telemedicine and Natural Disasters: Various Services, Requirements, Challenges, and General Framework," *Frontiers in Emergency Medicine*, 2022.
[8] Y. Emük *et al.*, "Telerehabilitation: A Promising Solution for Postearthquake Rehabilitation," İzmir Katip Çelebi Üniversitesi Sağlık Bilimleri Fakültesi Dergisi, vol. 8, no. 2, 2023, pp. 647–51.
[9] M. Nejadshafiee *et al*., "Providing Telenursing Care for Victims: A Simulated Study for Introducing of Possibility Nursing Interventions in Disasters," *BMC Medical Informatics and Decision Making*, vol. 22, no. 1, 2022, p. 54.
[10] J. Baker *et al*., "Telemedicine Technology: A Review of Services, Equipment, and Other Aspects," *Current Allergy and Asthma Reports*, vol. 18, no. 60, 2018.
[11] M. Khan *et al*., "Development of Smart E-Health System for Covid-19 Pandemic," *Proc. 23rd Int'l. Conf. Computer and Information Technology*, 2020, pp. 1–6.
[12] H. B. Azhar *et al*., "Progressive Web App for Real-Time Doctor-Patient Communication and Searchable Health Conditions," *Proc. E-Health and Bioengineering Conf*., 2022, pp. 1–5.
[13] F. García *et al.*, "Health-5G: A Mixed Reality-Based System for Remote Medical Assistance in Emergency Situations," *IEEE Access*, vol. 11, 2023.
[14] R. Thenmozhi *et al*., "Cloud-Enhanced Tele-Ent: A Scalable and Secure Ai-Driven Diagnostics for Remote Ear, Nose, and Throat Consultations," *Proc. Int'l. Conf. Artificial Intelligence for Innovations in Healthcare Industries*, vol. 1, 2023, pp. 1–6.
[15] Janus WebRTC server, https://janus.conf.meetecho.com/; accessed July, 2024.

Biographies

Sujit Kumar Sahu is presently a postdoctoral researcher at the Sant'Anna School of Advanced Studies in Pisa, Italy, where he is focusing on the development of a smart helmet for telemedicine applications. Prior to this role, he earned a dual Ph.D. degree in Robotics from the Sant'Anna School of Advanced Studies in Pisa, Italy and the University of Strasbourg, France. His research expertise encompasses a range of areas, including surgical robots, telemedicine, human-robot interaction, sensors, and actuators.

Molka Gharbaoui is an Assistant Professor at the Sant'Anna School of Advanced Studies in Pisa, Italy. She received her Ph.D. degree in Innovative Technologies of Information & Communications Engineering and Robotics in 2012 from the Sant'Anna School of Advanced Studies. Her main research interests are in the field of software-defined networking, network function virtualization, network orchestration, the development and the implementation of service-oriented architectures and service management for smart cities.

Anna Lina Ruscelli is an Assistant Professor at Scuola Superiore Sant'Anna, Italy. Her research activity spans from Quality of Service support for wireless networks, MAC protocols and scheduling algorithms, industrial wireless networks, telemedicine applications, ERTMS-ETCS railways systems and communications. She has more than fifteen years of experience in systems analysis and design, evaluation and testing of telecommunications networks acquired in the context of national and international projects and industrial collaborations with major public and private companies (Italian Railway Company RFI, Trenitalia, Shindler, National Institute for Workplace Accident Insurance-INAIL).

Gabriele Cecchetti is a researcher at TeCIP Institute of Scuola Superiore Sant'Anna, Pisa, Italy. He has been involved in various national and European FP7 and H2020 projects and was the Technical Leader of European project Shift2Rail H2020 OPTIMA about communication platform for traffic management demonstrator for rail systems. He is a computer engineer, and his research interests range from Quality-of-Service support over heterogeneous networks, IEEE 802.11e networks, scheduling algorithms for resource reservation, wireless link emulation, ERMTS-ETCS railway systems, railways signaling systems and communications, smart medical systems, embedded systems, and information systems.

Andrea Sgambelluri is an Assistant Professor at the Sant'Anna School of Advanced Studies in Pisa, Italy. He received his Ph.D. degree in Emerging Digital Technologies in 2015 from the Sant'Anna School of Advanced Studies. His main research interests are in the field of control plane techniques for both packet and optical networks, including Software Defined Networking (SDN), network reliability, industrial ethernet, switching, segment routing application and telemetry. He has been involved in Italian and European research projects on next generation optical networks and innovative solution for the automated control and dynamic management of future transport networks.

Piero Castoldi [SM'12] is Full Professor in Telecommunications and Leader of the "Networks and Services" research area and Director of the TeCIP Institute of Scuola Superiore Sant'Anna, Pisa, Italy. He has been involved with various responsibilities in several national and EU FP7 and H2020 projects and he has managed several corporate-sponsored projects by the Italian Railway Company, Ericsson, Telecom Italia. His most recent research interests lie in the areas of optical network architectures, interconnection networks for Data Centers, networks for industrial applications, 5G networking and vertical applications. He is author of more than 500 technical papers published in international journals and international conference proceedings. He has also filed more than 20 patents and he has authored a book on multiuser detection for CDMA mobile terminals published by Artech House.

Open access funding provided by Scuola Superiore Sant'Anna within the CRUI CARE agreement.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/