ICONS 2024

The Nineteenth International Conference on Systems

ISBN: 978-1-68558-175-6

May 26 - 30, 2024

Barcelona, Spain

ICONS 2024 Editors

Christoph Knieke, Technische Universität Clausthal, Germany

Eugen Borcoci, National University of Science and Technology Politehnica Bucharest, Romania

ICONS 2024

Forward

The Nineteenth International Conference on Systems (ICONS 2024), held between May 26-30, 2024 in Barcelona, Spain, continued a series of events covering a broad spectrum of topics, including fundamentals on designing, implementing, testing, validating and maintaining various kinds of software and hardware systems.

In the last years, new system concepts have been promoted and partially embedded in new deployments. Anticipative systems, autonomic and autonomous systems, self-adapting systems, or on-demand systems are systems exposing advanced features. These features demand special requirements specification mechanisms, advanced behavioral design patterns, special interaction protocols, and flexible implementation platforms. Additionally, they require new monitoring and management paradigms, as self-protection, self-diagnosing, self-maintenance become core design features.

The design of application-oriented systems is driven by application-specific requirements that have a very large spectrum. Despite the adoption of uniform frameworks and system design methodologies supported by appropriate models and system specification languages, the deployment of application-oriented systems raises critical problems. Specific requirements in terms of scalability, real-time, security, performance, accuracy, distribution, and user interaction drive the design decisions and implementations.

This leads to the need for gathering application-specific knowledge and develop particular design and implementation skills that can be reused in developing similar systems.
Validation and verification of safety requirements for complex systems containing hardware, software and human subsystems must be considered from early design phases. There is a need for rigorous analysis on the role of people and process causing hazards within safety-related systems; however, these claims are often made without a rigorous analysis of the human factors involved. Accurate identification and implementation of safety requirements for all elements of a system, including people and procedures become crucial in complex and critical systems, especially in safety-related projects from the civil aviation, defense health, and transport sectors.

Fundamentals on safety-related systems concern both positive (desired properties) and negative (undesired properties) aspects. Safety requirements are expressed at the individual equipment level and at the operational-environment level. However, ambiguity in safety requirements may lead to reliable unsafe systems. Additionally, the distribution of safety requirements between people and machines makes difficult automated proofs of system safety. This is somehow obscured by the difficulty of applying formal techniques (usually used for equipment-related safety requirements) to derivation and satisfaction of human-related safety requirements (usually, human factors techniques are used).

We welcomed academic, research and industry contributions. The conference had the following tracks:

- Complex and specialized systems
- Embedded systems and applications/services
- Computer vision and computer graphics
- Application-oriented systems

We take here the opportunity to warmly thank all the members of the ICONS 2024 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to ICONS 2024. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also thank the members of the ICONS 2024 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that ICONS 2024 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the area of systems. We also hope that Barcelona provided a pleasant environment during the conference and everyone saved some time to enjoy the historic charm of the city.

ICONS 2024 Chairs

ICONS Steering Committee

David Inkermann, Technische Universität Clausthal, Institute of Mechanical Engineering, Germany
Christoph Knieke, Technische Universität Clausthal, Institute for Software and Systems Engineering, Germany
Giulio Telleschi, MBDA, Italy
Mo Mansouri, Stevens Institute of Technology, USA/ University of South-Eastern Norway, Norway
Mark Austin, University of Maryland at College Park, USA

ICONS Publicity Chairs

Laura Garcia, Universidad Politécnica de Cartagena, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

ICONS 2024

Committee

ICONS Steering Committee

David Inkermann, Technische Universität Clausthal, Institute of Mechanical Engineering, Germany
Christoph Knieke, Technische Universität Clausthal, Institute for Software and Systems Engineering, Germany
Giulio Telleschi, MBDA, Italy
Mo Mansouri, Stevens Institute of Technology, USA/ University of South-Eastern Norway, Norway
Mark Austin, University of Maryland at College Park, USA

ICONS 2024 Publicity Chairs

Laura Garcia, Universidad Politécnica de Cartagena, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

ICONS 2024 Technical Program Committee

Qammer H. Abbasi, University of Glasgow, Scotland, UK
Mohamed A. Abd El Ghany, German University in Cairo, Egypt
Witold Abramowicz, Poznan University of Economics, Poland
Afaq Ahmad, Sultan Qaboos University, Oman
Abdelouhab Aitouche, YNCREA/HEI | University of Lille, France
Ali Al-Humairi, German University of Technology (GUtech), Oman
Walid Al-Hussaibi, Southern Technical University (STU), Iraq
Saloua Bel Hadj Ali, University of Tunis-El Manar / University of Gabès, Tunisia
Fady Alnajjar, United Arab Emirates University, UAE
Yasmine Arafa, University of Greenwich - School of Computing and Mathematical Sciences, UK
Marc Austin, University Of Maryland, USA
Muhammed Ali Aydin, Istanbul University-Cerrahpasa, Turkey
Snježana Babić, Juraj Dobrila University of Pula, Croatia
Lubomir Bakule, Institute of Information Theory and Automation, Czech Republic
Walter Balzano, Università degli Studi di Napoli Federico II, Italy
Kamel Barkaoui, Cedric - Cnam, Paris, France
Janibul Bashir, National Institute of Technology, Srinagar, India
Suvadip Batabyal, BITS Pilani, Hyderabad Campus, India
Héla Belkhiria, Higher National Engineering School of Tunis - University of Tunis / Polytechnic School of Tunisia, Tunisia
Abir Ben Ali, National School of Computer Sciences (ENSI), Tunisia
Alejandro J. Bianchi, LIVEWARE S.A. / Universidad Catolica Argentina, Argentina
Francesco Bianconi, Università degli Studi di Perugia, Italy
Razvan Bocu, Transilvania University of Brasov, Romania
Birthe Boehm, Siemens AG, Germany
Sander Bohte, Machine Learning group - CWI, Amsterdam, The Netherlands

Tim O'Neil, University of Akron, USA
Maurizio Palesi, University of Catania, Italy
Francesca Palumbo, University of Cagliari, Italy
Samuel Pastva, Masaryk University in Brno, Czech Republic
Szczepan Paszkiel, Opole University of Technology, Poland
Davide Patti, University of Catania, Italy
George Perry, University of Texas at San Antonio, USA
Valerio Persico, University of Napoli "Federico II", Italy
Safanah M. Raafat, University of Technology, Baghdad, Iraq
Sujan Rajbhandari, Coventry University, UK
Ramakrishnan Raman, Honeywell Technology Solutions, Bangalore, India
Grzegorz Redlarski, Gdańsk University of Technology, Poland
Piotr Remlein, Poznan University of Technology, Poland
Bernard Riera, Université de Reims Champagne-Ardenne, France
Javier Rocher, Universitat Politecnica de Valencia, Spain
Juha Röning, University of Oulu, Finland
Somayeh Sadeghi-Kohan, Paderborn University, Germany
Francesca Saglietti, University of Erlangen-Nuremberg, Germany
Souhir Sallem, National School of Engineering of Sfax, Tunisia
Areeg Samir, The Arctic University of Norway, Norway
Christophe Sauvey, Universite de Lorraine, France
Tomas Schweigert, Expleo, Germany
Hayat Semlali, Cadi Ayyad University, Morocco
Avi Shaked, Tel Aviv University, Israel
Yilun Shang, Northumbria University, UK
Charlie Y. Shim, Kutztown University of Pennsylvania, USA
Yong-Sang Shim, Kutztown University of Pennsylvania, USA
Tajinder Singh, Sant Longowal Institute of Engineering & Technology, India
Seyit Ahmet Sis, Balikesir University / BİLGEM-TÜBİTAK (The Scientific and Technological Research Council of Turkey), Turkey
Rocky Slavin, University of Texas at San Antonio, USA
Pedro Sousa, University of Minho, Portugal
Olarik Surinta, Mahasarakham University, Thailand
Lo'ai A. Tawalbeh, Texas A&M University-San Antonio, USA
Shahab Tayeb, California State University, USA
Bedir Tekinerdogan, Wageningen University, Netherlands
Massimo Torquati, University of Pisa, Italy
Ahmed Toumi, Sfax University, Tunisia
Carlos M. Travieso-González, University of Las Palmas de Gran Canaria (ULPGC), Spain
Denis Trček, University of Ljubljana, Slovenia
Ruthvik Vaila, Bastian Solutions (Toyota Advanced Logistics), Boise, USA
Penka Valkova Georgieva, Burgas Free University, Bulgaria
Irena Valova, University of Ruse, Bulgaria
Szilvia Váradi, University of Szeged, Hungary
Weier Wan, Stanford University, USA
Wenxi Wang, University of Texas at Austin, USA
Natalia Wawrzyniak, Maritime University of Szczecin, Poland
Katarzyna Wegrzyn-Wolska, AlliansTIC Laboratory | EFREI PARIS, France

Copyright Information

Table of Contents

Air Quality Monitoring Platform for Virtual Reality-Enabled Digital Twin: the Use Case of Cartagena (Spain)

Laura García
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: laura.garcia@upct.es

Daniel Montoya-Garcia
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: daniel.montoya@edu.upct.es

Francisco Sanchez-Rubio
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: francisco.sanchez@edu.upct.es

Antonio-Javier Garcia-Sanchez
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: antoniojavier.garcia@upct.es

Rafael Asorey-Cacheda
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: rafael.asorey@upct.es

Joan Garcia-Haro
Deparment of Information and Communications Technologies
Universidad Politécnica de Cartagena
Cartagena, Spain
email: joang.haro@upct.es

***Abstract*—Recent advances in the virtualization of the world we live in have enabled an increasing number of new functionalities that are generating increasing interest from governments, private organizations, and the general public. One of these functionalities is the real-time display of Internet of Things (IoT) data in different types of environments and at varying scales. From maps encompassing wide regions, to buildings, and objects, such as industrial machinery. Air quality monitoring is one of the most popular uses of IoT in Smart Cities due to the severe health effects that air pollution may cause in people. As such, there is a growing concern in creating new tools to enhance the accessibility of the data and increase the awareness regarding air quality. This paper addresses this specific matter, presenting a Virtual Reality-enabled Digital Twin for air quality monitoring platforms. The use case reported in this work applies to the city of Cartagena (Spain), where several of our air quality monitoring devices for polluting gases and suspended particulate matter are deployed. The digital twin was developed using Unity and Citigen for environment development coupled with the data stored in the servers localized at the Universidad Politécnica de Cartagena (UPCT) from our LoRaWAN air quality monitoring IoT network.**

Keywords-Digital Twin; air quality; Virtual Reality; Unity.

I. Introduction

In the recent years, IoT has been adopted for multiple purposes, accounting for more than 9 billion devices in the world [1]. Smart cities are one of the applications where a higher number of devices is needed, as it is necessary for characterizing each area of the city. Specifically, air quality monitoring devices are present in most smart cities due to the growing concern regarding the pollution generated by increasing amounts of traffic. These devices are often equipped with sensors that measure two main types of pollutants: gases and suspended particulate matter. The most relevant polluting gases include CO, SO_2, O_3. and NO_2 [2]. Conversely, suspended particulate matter is measured for three main particulate sizes, including PM_1, $PM_{2.5}$, and PM_{10}. Monitoring these pollutants in real-time leads to a a great amount of data that needs to be managed and presented to the users in an accessible manner. Dashboards allow data representation in graphical and table formats, but it is limited in functionality. For that reason, new forms of data representation leveraging virtual reality and augmented reality are being considered [3].

Digital Twins provide new possibilities for data representation enabling the creation of digital replicas of the physical world that can be used for real-time monitoring [4]. Other functionalities that can be included in digital twins are learning, predictions, and simulations [5]. Most Digital Twins currently in use have been developed for engineering applications that includes a series of objects or spaces that usually do not surpass the scale of a small number of buildings. But as this technology evolves, the interest of using it for more ambitious purposes has increased. Smart-cities are viewing digital twins as the next tool for providing smarter and more flexible services [6]. However, the creation of a city-level digital twin introduces new challenges as the functional boundaries of a city have higher complexity, not allowing a straightforward upscale.

The technology used to develop a digital twin depends on its intended use. Digital twins intended for buildings are often created based on technologies, such as Building Information Modelling (BIM) and Computer-Aided Design (CAD) that have evolved from its first stages as software to aid builders in the construction process [7]. However, these software tools are not intended for large scale city-like representations. Therefore, it is necessary to use alternative tools, such as video game engines, which have substantially evolved allowing the creation of meta-worlds that can be interacted with. In this paper, we present a digital twin of a smart city for air quality monitoring enabled by virtual-reality. It was developed using Unity and the Citigen tool for the creation of a 3D map of the city of Cartagena (Spain). Each air quality monitoring device deployed in the city had its corresponding object in Unity that could be selected to display a dashboard with the current air pollution readings gathered from the sensors. The solution is enabled by virtual reality, facilitating the navigation of the city.

The rest of the paper is organized as follows, Section 2 presents the related work. The methodology is presented in Section 3. The results are discussed in Section 4. Lastly, the conclusion and future work is presented in Section 5.

II. Related Work

Many of the currently existing digital twins focus on buildings and monitoring the physical factors that affect them or the people inhabiting them. One of them is the digital twin created by Khajavi *et al.* for the facade of an office building [8]. It was equipped with six sensor devices for real-time monitoring of temperature, relative humidity, and environmental lighting. More than 25,000 readings where stored, analyzed and represented in the limited digital twin that overlapped a 2D graphical representation of the lighting received by the sensors using different shades of color on top of an image of the building's facade. However, city-scale digital twins have begun to be developed as well. Mohammadi and Taylor presented a paradigm for a smart city digital twin [9]. Their use case for the city of Atlanta was developed in Unity and based on virtual reality. A plugin was developed to provide analytics according to the data. Furthermore, an augmented reality crowd-sourcing module was included as a tagging system, using a mobile application to run it. In addition, Wolf *et al.* presented a digital twin for emergency management in smart cities [10]. The digital twin includes information, such as weather and traffic data for improved service coordination. The digital twin was developed using the Microsoft Azure cloud for its analytical functionalities, and the maps were generated utilizing TomTom and the Microsoft Azure Maps Web Software Development Kit.

As air quality monitoring has been gaining more relevance, the prospect of creating digital twins that feature air quality data has been increasing. Some of the proposals focused on indoor environments. Qian *et al.* presented an indoor air quality intelligent management approach based on digital twin platforms [4]. Data from different sources was collected, integrated, analyzed, and displayed through a web-based interface. The system was tested in a traditional home in china. The results showed that winter morning presented the most air quality problems. The definition of healthy quality thresholds allowed identifying air quality concerns that could be acted on, solving these problems within 30 minutes since their detection. Furthermore, Ariswala *et al.* used BIM to create a digital twin for monitoring and control of equivalent carbon dioxide in indoor spaces [11]. The solution used IoT devices equipped with low cost sensors, the Microsoft Azure Cloud Platform, Azure Digital Twins, Azure Machine Learning, Power Bi, and the Polycam application to scan the desired building. The resulting digital twin with the 3D implementation of the building included a dashboard to display real-time data and forecasts.

Smart cities have a great interest in air quality monitoring as well. As such, city-scale digital twins are also including these features. Ariansyah *et al.* presented a digital twin for air quality monitoring in smart cities using mix reality technology [12]. The proposal integrated air quality, meteorological, urban infrastructure, and traffic pattern data that could be displayed through a mixed reality headset that represented the map and information as an object placed in the real world. Lastly, Siddaraju *et al.* proposed a digital twin for PM2.5 estimation [14]. This version of digital twin was focused on the simulation of the environment to determine PM2.5 levels and did not include a 3D digital representation of the location where the meteorology stations were deployed. The authors used machine-learning techniques to implement PM2.5 prediction models.

Instead of focusing on building-scale digital twins or 2D, visually limited representations, this paper presents the implementation of a city-scale 3D digital twin for outdoor air quality monitoring enabled by virtual reality.

III. Methodology

In this section, the methodology followed to create the Air Quality monitoring system using virtual reality is presented.

A. Architecture

The architecture of the proposed integral solution for an air quality monitoring platform enabled by a virtual reality-based digital twin is presented in Figure 1. IoT monitoring devices for polluting gases and suspended particulate matter with different sizes are deployed in the city of Cartagena (Spain). These devices take measures periodically and send them to the LoRaWAN Gateway. The Gateway communicates to the LoRaWAN server through MQTT protocol. Telegraf is subscribed to the MQTT topics belonging to the IoT devices to capture the data, provide context and store it at the Influx database. The 3D environment of the Digital Twin for air quality monitoring is created using Citygen3D, a tool for map creations in Unity. The panels that display the device's data are created as objects, and the data from the database is accessed by Unity to display it. Lastly, the users can experience the Digital Twin with a VR headset or in an ordinary monitor.

The following subsections will detail the implementation of each of the elements of the architecture.

B. Air quality monitoring sensor devices

Our air pollution monitoring devices were developed independently, one encompassing the polluting gas sensors (SO_2,

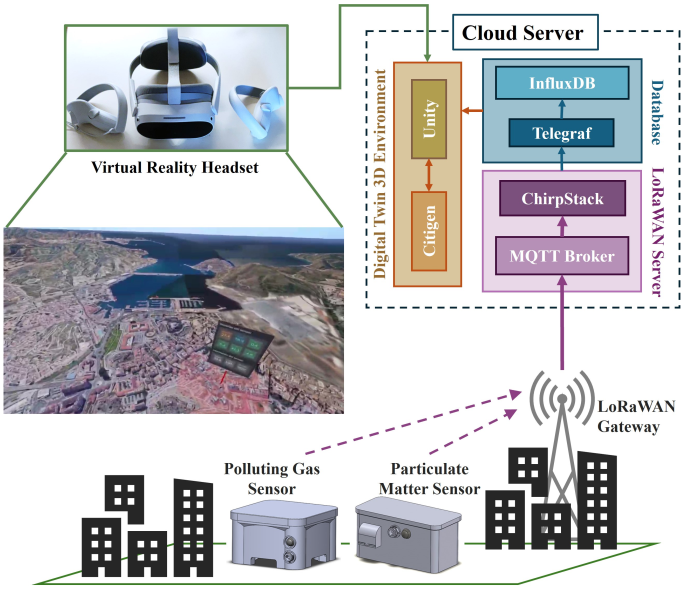

Figure 1. Architecture of the proposed VR-enabled Digital Twin for air quality monitoring.

NO_2, O_3 and CO), and one including the suspended particulate matter sensor (PM_1, $PM_{2.5}$, and PM_{10}) [2]. Both devices include a Pycom embedded system that manages data collection, message formatting, and data transmission. The three devices deployed in Cartagena were equipped with a LoRaWAN transceiver for long range communications. Figures 2, 3, and 4 present the data gathered by the suspended particulate matter for a common weekday. As it can be seen, the concentration of suspended particulate matter increases with rush hour, particularly PM10. The polluting gas data is presented in Figures 5, 6, 7, and 8, respectively. It can be seen that the gases present a different behaviour compared to each other and the suspended particles. Newly deployed devices are easily integrated in the solution, only requiring their activation in the LoRaWAN network. If necessary, other sensors intended for different air quality metrics could be similarly embedded to be adapted to different environments and purposes.

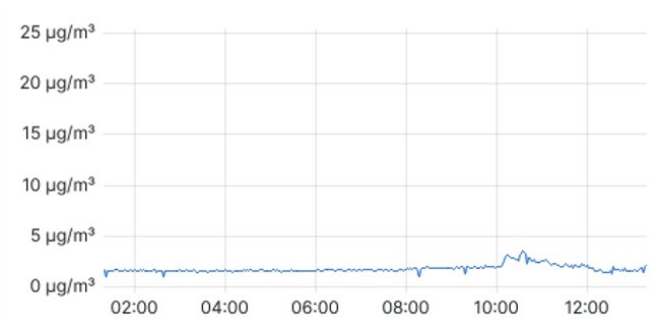

Figure 2. PM_1 data from suspended particulate matter device.

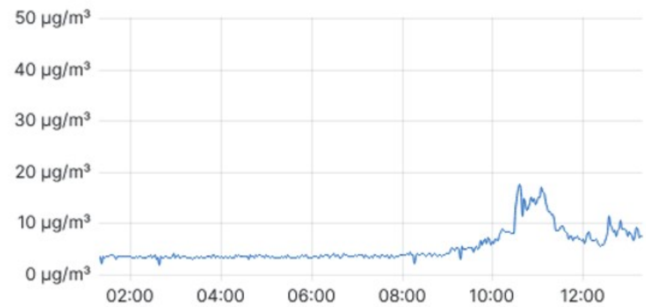

Figure 3. $PM_{2.5}$data from suspended particulate matter device.

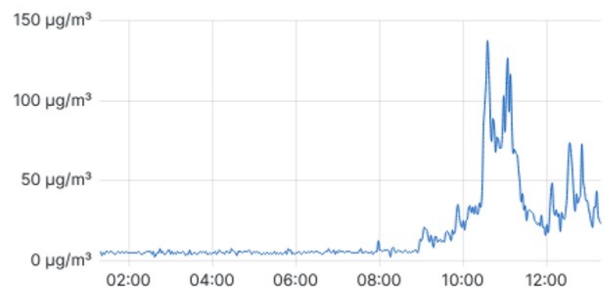

Figure 4. PM_{10}data from suspended particulate matter device.

C. Virtual Reality Headset

Virtual Reality Headsets are devices that show images generated by a computer, usually in 3D, through screens located very close to the eyes. The headsets occupy all the user's vision field, providing an immersive experience. In this project, two types of VR Headsets were used to assess user experience. No significant differences were found in their use. The characteristics of the headsets are the following:

1) PicoBlaze: The Pico 4 VR Headset [14] includes a Snapdragon XR2 processor which can reach speeds of up to 2.84 GHz. It offers 8 GB of RAM, ensuring smooth performance in applications and games. The employed model offers 128 GB of storage. Regarding connectivity, it supports WiFi 6 (802.11 a/b/g/n/ac/ax) and 2x2 MIMO dual-band (2.4 GHz/5 GHz). It also has Bluetooth 5.1 for pairing with other devices. The display is high quality, with two 2.56-inch LCD panels and a resolution of 2,160 x 2,160 pixels per eye. It offers a high pixel density (PPI: 1200), a refresh rate of 72/90 Hz and a wide 105° field of view. In addition, the interpupillary distance adjustment is electric and varies between 62-72 mm, adjusting it correctly allows you to view images without distortion. It features dual stereo speakers and a dual microphone for an immersive experience. Echo cancellation technology with 50 dB reduction ensures clear communication. This device runs on the Pico OS 5.0 operating system.

2) Quest 2: The Quest 2 VR headset [15] also includes a Snapdragon XR2 processor. Regarding storage, our headset is equipped with 256 GB. The connectivity of this device has support for WiFi 6 (802.11 a/b/g/n/ac/ax) and 2x2 MIMO dual-band (2.4 GHz/5 GHz). It also has Oculus Link, to connect to

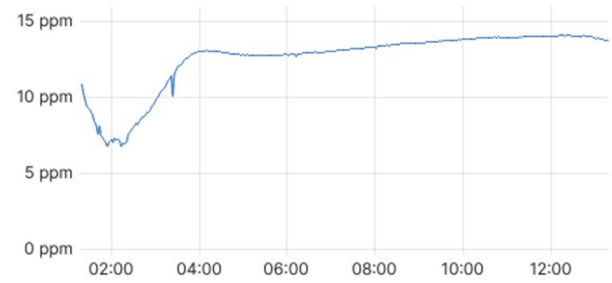

Figure 5. COdata from pollutant gas monitoring device.

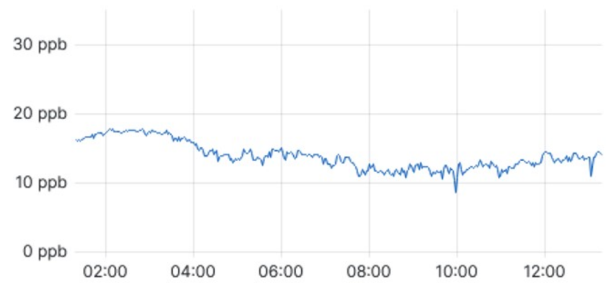

Figure 6. SO_2data from pollutant gas monitoring device.

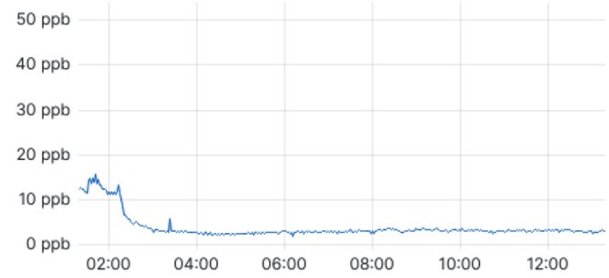

Figure 7. O_3data from pollutant gas monitoring device.

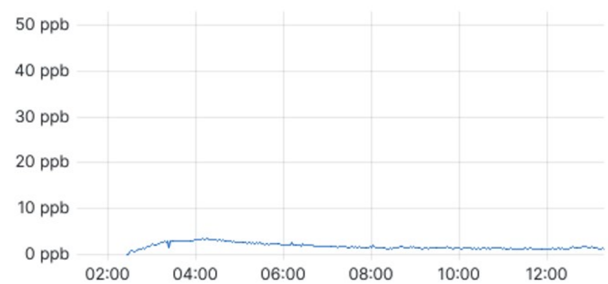

Figure 8. NO_2data from pollutant gas monitoring device.

a PC via USB cable. The display has two LCD panels with a resolution of 3664 x 1920 pixels per eye. It offers a high pixel density (PPI: 1100), a refresh rate of 72/90 Hz and a wide field of view of 105°. The electric interpupillary distance adjustment varies between 63-72 mm. It features dual stereo speakers and a 3.5mm audio port for external headphones. The echo cancellation provides a 45 dB reduction. This device runs on the Oculus OS operating system, which includes a stable and efficient Android-based platform.

D. Unity

The digital twin needs to be supported by a powerful software that enables smooth visualization of 3D environments and assets. Furthermore, it is necessary to ensure its compatibility with popular VR headsets to provide accessibility. Unity [16] is the most popular game development engine and 3D application creation platform, and thus, it has an extensive community and substantial development tools and support. It is used by developers to create games, interactive simulations, virtual reality (VR) and augmented reality experiences, and 2D and 3D applications. Unity is known for its versatility and ease of use, making it perfect for the development of this project. It allows importing and managing 3D models, textures, sounds and animations in order to speed up the development process. It can also be programmed in languages, such as C# and JavaScript to create custom behaviors. The physics system that allows simulating realistic movements and collisions in the virtual world.

Some of the main aspects to be considered when creating the 3D map of the city of Cartagena included the need of changing between predetermined levels of detail depending on the distance of the object, to increase or reduce the number of polygons to optimize the scene. The textures are also critical. A height map was used instead of a normal map to take into account the angle of the viewer with the surface. Its downside is the longer processing times. Texture compression improved these times, selecting ASTC for our project. For lighting, the use of lights was reduced as mush as possible, utilizing baked lights whenever possible. These type of lights reduce the computational cost of rendering the scene, but using many of them increase the number of objects and thus the processing time. All options for improved visual quality could not be included and objects had to be configured as static to avoid increases in execution time.

Other configurations include animations, physics, object loading, and geometry. Animation changes are not problematic, but they have a significant performance cost, so the blend nodes were kept below 6 to reduce this cost. Each object needs to be linked to an animator object, but interpolators are recommended instead of animators. They can be implemented using custom scripts, especially for the user interface. Regarding physics, the reuse of collision callbacks was enabled. Combined primitive colliders were employed to mimic the object they are attached to. Moreover, the layer-based collision detection was used to detect collisions of an object of a predetermined layer with the rest of the objects. By default, Unity loads objects on top of other objects which causes overlapping pixels and longer execution time. Using Occlusion Culling prevents Unity from loading objects outside the camera view. However, it must be done with each object, which is time consuming for an entire city plan. Lastly, the

 ISBN: 978-1-68558-175-6

wireframe mode is used to optimize the geometry. It makes the level of detail of the objects depend on the distance to the view. The models must be created manually, adding the LOD Group component, and placing them in the renderers section.

In order to display the information received from the sensors, a series of objects are included to display the individual information of each sensor. To do this, an object is added at the position where the sensor we want to display is located. This object has an associated script that opens a screen to view the information when clicking on it. Other objects where added inside, since the first object only serves to open the screen. The next object is where the information is printed. It displays the sensor measurements and is placed on the previous object. The object is comprised of several sections including temperature, humidity, CO, NO2, O3 and SO2 (See Figure 9).

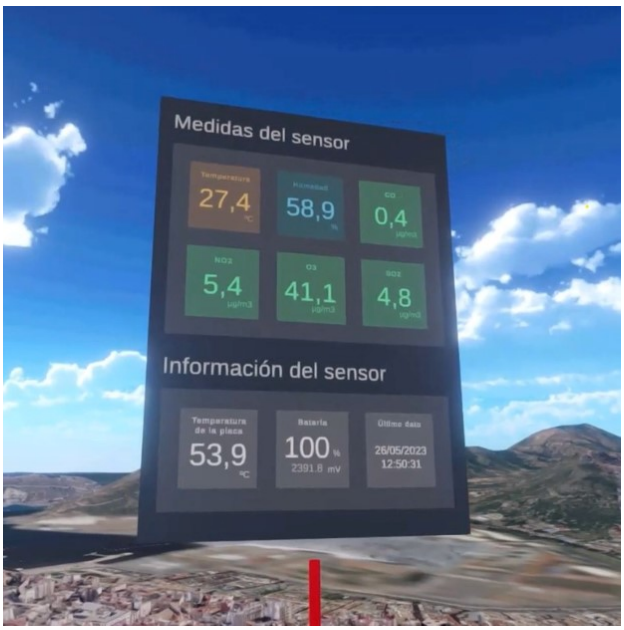

Figure 9. Unity object displaying air quality data from sensor devices.

E. Citygen3D

CityGen3D [17] is an extension for the Unity editor designed to simplify the automated creation of three-dimensional scenarios based on real-world map data from OpenStreetMap. CityGen3D does not require additional coding. A specific location, by means of latitude and longitude coordinates, can be provided and it will download and analyze the real data of the world map inside Unity to generate the scene. To make it a more realistic experience, CityGen3D adds heightmaps in order to be able to appreciate the height differences (mountains, ports, etc). This presented some problems because the height difference in the relief generated too many triangles to represent it correctly. Therefore, modifications were necessary to prevent the application from excessive loading times. Splat textures could be selected from different types of terrain (grass, asphalt and earth among others) to make the world more realistic. The road network could also be generated automatically using the combination of Unity and Citygen3D. Most trees were eliminated since they occupied a great part of the rendering memory and were unnecessary information for our purpose.

F. InfluxDB

The database used in our project is InfluxDB [18]. It is a time series database that can receive and process a large amount of data thanks to the TSM engine, which guarantees data availability, integrity and retrieval.

IV. Results

This section presents the results of our implementation of complete virtual reality-enabled digital twin of the city of Cartagena, that displays the data gathered by all the air quality monitoring devices deployed in the city.

The result for the creation of a virtual world representing Cartagena using the Citygen3D tool is shown in Figure 10. This specialized software has demonstrated its ability to generate detailed three-dimensional environments with high levels of realism.

Figure 10. Empty map of Cartagena using Citigen3D.

The final visual presentation inside the virtual reality glasses can be seen in Figure 11. The differences in height are clearly appreciable and the object with the dashboard for air quality data representation is easily detectable.

Figure 11. Complete visualization of the digital twin using the VR headset.

The users can interact with the panels using the controllers associated with the virtual reality glasses, as shown in Figure 12.

We have therefore succeeded in the implementation of the digital twin of Cartagena. It can be navigated seamlessly with the CR headset and the panels can be activated and removed at will. Other existing works on digital twins for air quality monitoring use 2D representations [8], [14], are intended for

Figure 12. Interaction of VR controllers with dashboard objects.

other parts of the Reality–virtuality spectrum [9], [12], or target only one building [4], [11]. The digital twin presented in this paper encompasses the city of Cartagena (Spain) and provides its 3D representation enabled by virtual reality.

V. CONCLUSION AND FUTURE WORK

Digital twin technology is currently evolving, adding more functionalities and scaling to bigger environments. Smart cities are viewing this technology as a tool to provide accessible information to its citizens. One of the main aspects to transfer into digital twins is air quality monitoring IoT solutions. In this paper, we describe the implementation of our digital twin for air quality monitoring in the city of Cartagena (Spain), which can display the data gathered from polluting gas and suspended particulate matter sensors at different locations of the digital twin map. It is enhanced by virtual reality, providing an immersive experience for the user. There is however ongoing challenges to be addressed. The main one being the optimization of polygons that comprise the city 3D mesh and the trade-off between visualization quality and performance. Nevertheless, the growing interest in this technologies has led to conversations with several institutions, where the identified applications of the digital twin were not limited to data visualization, but also allows for helping policy-makers in performing informed decisions on redirecting traffic flows and urban planning.

For future work, we will implement Artificial Intelligence-based solutions that evaluate air quality form the sensors deployed in the city and present the predictions as heat-maps over the city, allowing the user to quickly determine the areas of the city with more expected air pollution. This also implies spatial predictions to obtain detailed information even in areas without air quality monitoring devices.

ACKNOWLEDGMENT

This work was supported by the grants PID2020-116329GB-C22, TED2021-129336B-I00, and FJC2021-047073-I, funded by MCIN/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR. This work was also funded by Fundación Séneca (22236/PDC/23). This work was also a result of the ThinkInAzul and AgroAlNext programmes, supported by MICIU with funding from European Union NextGenerationEU (PRTR-C17.I1) and by Fundación Séneca with funding from Comunidad Autónoma Región de Murcia (CARM).

REFERENCES

[1] IoT Cybersecurity: Protecting the merging of the physical and digital world. [Online] Available: https://www.telefonica.com/en/communication-room/blog/iot-cybersecurity-protecting-the-merging-of-the-physical-and-digital-world/ (Accessed: 2024-3-3)

[2] L. García *et al.*, "Smart air quality monitoring IoT-based infrastructure for industrial environments", Sensors, vol. 22, no. 23, pp. 9221, 2022.

[3] D. Topping *et al.*, "Digital twins of urban air quality: Opportunities and challenges", Frontiers in Sustainable Cities, vol. 3, pp. 786563, 2021.

[4] Y. Qian, J. Leng, K. Zhou, and Y. Liu, "How to measure and control indoor air quality based on intelligent digital twin platforms: A case study in China", Building and Environment, pp. 111349, 2024.

[5] Z. Wang *et al.*, "Mobility digital twin: Concept, architecture, case study, and future challenges", IEEE Internet of Things Journal, vol. 9, no. 18, pp. 17452–17467, 2022.

[6] L. Wan, T. Nochata, and J. M. Schooling, "Developing a city-level digital twin–propositions and a case study", in International conference on smart infrastructure and construction 2019 (ICSIC) driving data-informed decision-making, Cambridge, UK, 8-10 July 2019, pp. 187–194.

[7] S. HongHong, Y. Gang, L. Haijiang, Z. Tian, J. Annan, "Digital twin enhanced BIM to shape full life cycle digital transformation for bridge engineering", vol. 147, pp. 104736, 2023.

[8] S. H. Khajavi, N. H. Motlagh, A. Jaribion, L. Werner, and J. Holmström, "Digital twin: vision, benefits, boundaries, and creation for buildings", IEEE Access, vol. 7, pp. 147406–147419, 2019.

[9] N. Mohammadi and J. E. Taylor, "Smart city digital twins", in 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November - 1 December 2017, pp. 1–5.

[10] K. Wolf, R. J. Dawson, J. P. Mills, P. Blythe, and J. Morley, "Towards a digital twin for supporting multi-agency incident management in a smart city", Scientific reports, vol. 12, no. 1, pp. 16221, 2022.

[11] A. Ariswala, F. Elghaish, and M. Zoher, "Digital twin with Machine learning for predictive monitoring of CO2 equivalent from existing buildings", Energy and Buildings, vol. 284, pp. 112851, 2023.

[12] D. Ariansyah, M. Isnan, R. Rahutomo, and B. Pardamean, "Digital Twin (DT) Smart City for Air Quality Management", Procedia Computer Science, vol. 227, pp. 524–533, 2023.

[13] P. M. Siddaraju, S. Sameer, S. S. Manjunath and Sneha, "Digital twin for image-based particle pollutant matter prognosis", Journal of The Institution of Engineers (India): Series B, vol. 104, no. 2, pp. 351–357, 2023.

[14] Pico 4 specifications [Online] Available: https://www.picoxr.com/es/products/pico4/specs (Accessed: 2024-4-8)

[15] Oculus Quest 2 Specifications [Online] https://studiox.lib.rochester.edu/oculus-quest-2-specifications/

[16] Unity [Online] Available: https://assetstore.unity.com/ (Accessed: 2024-2-26)

[17] CityGen3D [Online] Available: https://www.citygen3d.com/ (Accessed: 2023-12-26)

[18] InfluxDB 1.1 documentation [Online] Available: https://docs.influxdata.com/influxdb/v2.0/get-started/ (Accessed: 2024-2-26)

Virtual Sensor Simulation and Post-Processing on the Hardware-in-the-Loop Test System for Autonomous Vehicles in the Cyber-Physical Traffic System

Taihao Li
Scientific assistant
Ostfalia University of Applied Sciences
Wolfenbüttel, Germany
email: ta.li@ostfalia.de

Marian Göllner
Scientific assistant
Ostfalia University of Applied Sciences
Wolfenbüttel, Germany
email: mar.goellner@ostfalia.de

Sven Jacobitz
Scientific assistant
Ostfalia University of Applied Sciences
Wolfenbüttel, Germany
email: sve.jacobitz@ostfalia.de

Xiaobo Liu-Henke
Professor
Ostfalia University of Applied Sciences
Wolfenbüttel, Germany
email: x.liu-henke@ostfalia.de

***Abstract*— Automated driving and connected cyber-physical traffic systems present increasing challenges for the development and validation of advanced driver assistance systems and automated driving functions. In particular, real-time optimization and testing involves significant workload and risk. Providing a holistic, flexibly configurable testbed with real-time capabilities for the entire vehicle can solve this problem. However, in order to make the functionality more accurately verified by the test bench, sensor simulation is an important component, i.e., the ability to generate real sensor information in a simulated environment. In addition, the data structure of the virtual sensor, as well as the transmitting type and sampling frequency, should be close to or even consistent with that of its real sensor. In addition, we also add the noise from the real sensor to the virtual sensor. The referenced noise values are taken from the data sheet of the real sensor. This alignment enables the test bench to better test the real-time functionality of the vehicle and its ability to process the sensor signals.**

Keywords—Cyber-Physical System; Virturl sensor; Autonomous Driving; Post-Processing; Real-Time Testing

I. INTRODUCTION

Mobility is undergoing disruptive change due to the increasing digitization and networking of vehicles. The autonomous driving of electric hybrid vehicles in highly interconnected Cyber-Physical traffic systems (CPS) is one of the core technologies in this digital transformation process. The variety of applications for autonomous vehicles requires ever more diverse sensor technology, as well as ever more complex and intelligent algorithms from the fields of modern control technology and Artificial Intelligence (AI). This results in increasingly complex systems. Not only because of the increasing range of functionality, but also because of the constantly growing degree of interconnection [1].

The development of such systems is closely linked to safety engineering requirements and is therefore highly complex. In order to study the integrated overall functionality of intelligent vehicles that are capable of autonomous driving, a complete vehicle test bed is essential. This test bed should accurately represent the complete system of road, connected vehicles, and connected driving environment, as well as stimulate the vehicle's sensors. For the above reasons presented the paper [2] the concept of a holistic, highly flexibly configurable real-time test system for intelligent vehicles in cooperating cyber-physical traffic systems, called ERAGON. In a closed loop together with the function carrier AURONA, this system is able to simulate and stimulate the entire autonomous vehicle system, starting from the infusion of raw sensor data via the development and testing of AI functions up to the stimulation of realistic driving situations. Therefore, it is particularly important to test the function under test with a simulation test bed for sensor data. It is ideal for the simulated data to be bit-for-bit identical to the data generated by the physical sensor in the real-world scenario. However, this level of similarity may not be necessary or achievable. The aim of sensor simulation is to achieve a level of data fidelity that ensures the decisions made by the control algorithms are equivalent to those made in reality. Therefore, the focus should be on achieving a high level of data fidelity [3].

This paper aims to discuss the virtual sensor simulation in ERAGON, how they are transmitted, and the required post-processing and applications. The rest of the paper is organized as follows. Section II presents the holistic model-based RCP methodology for the development of complex, interconnected mechatronic systems. Section III presents state of the arts of the sensor simulation. In Section IV, the concept of the simulation of the virtual sensor and their post-processing will be introduced. Finally, Section V provides a summary of the contents as well as an outlook on further work.

II. METHODOLOGY

Through the methodology of mechatronic development, specifically mechatronic structuring, the interconnected CPS is divided into hierarchically organized subsystems across four levels of hierarchy: mechatronic functional modules (MFM), mechatronic functional groups (MFG), and autonomous mechatronic systems (AMS), and Networked mechatronic system (NMS) [4]. The outcome of this structuring is a functional decomposition of the entire system into encapsulated modules. These modules are arranged hierarchically and have clearly defined physical and informational interfaces in both horizontal and vertical directions, which lays the foundation for later integration into the overall system [5].

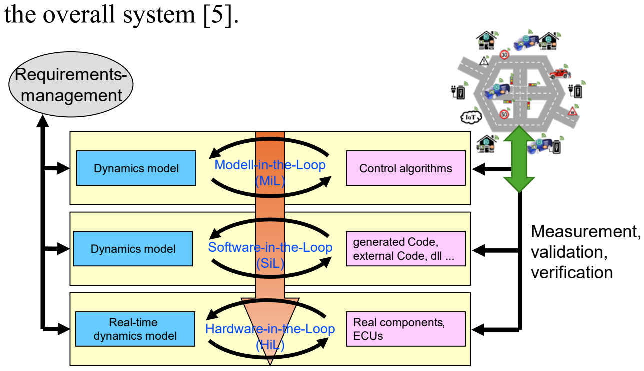

Figure 1. Seamless model-based development and validation process[6]

After establishing a hierarchical structure and specifying all interfaces, a model-based, function-centric approach is used to the design of each discrete module, adopting a bottom-up strategy. The initiation occurs at the most fundamental and critical level, the MFM, progressing upwards through the hierarchy to assimilate these modules into more complex functions. The integration and combination of these functions within the larger framework (mechatronic assembly) leverage the model-based mechatronic development cycle. Subsequently, Model-in-the-Loop (MiL) simulations are conducted to create and refine control algorithms and artificial intelligence based on a virtual or mathematical model, which are then trialed using a vehicle simulation. Within the Software-in-the-Loop (SiL) phase, these algorithms, once validated through simulation, are translated into operational code via automatic code generation, then assessed offline on a virtual platform. The sequence advances to Hardware-in-the-Loop (HiL) simulations, utilizing an augmented real-time vehicle model integrated with physical components for online verification and enhancement of the algorithms and smart functions under actual operational conditions [7]. This procedural sequence is delineated in Figure 1.

In this paper where the vehicle under test belongs to AMS due to the autonomous driving function it has, and its other underlying actuators such as motor braking are MFM. the function of assisted driving is MFG. In testing it is a HiL test because the object under test is an entity with physical components.

III. State Of Arts

Efficient functional verification is a significant challenge in realizing autonomous driving, as stated in the literature [8]. It is crucial to ensure that the functions designed in the overall system are verifiably safe in terms of output quality and the probability of misinterpretation [9]. It is necessary to repeat the tests for as many situations as possible that the vehicle may encounter. Hundreds of millions of kilometers of testing are necessary if this task is to be accomplished under real road traffic conditions [10]. Reliable and robust environment sensing through camera, RADAR, and LiDAR sensors is a key element of Advanced Driver Assistance Systems (ADAS) and Autonomous Driving (AD-assisted systems). Synthetic sensor data is required for driving simulations to develop and validate sensor-based algorithms. The classification of automotive sensing sensor models is based on their modeling approach and coverage effects, and can be divided into three categories: ideal, phenomenological, and physical models [11].

The Ideal Sensor Model, alternatively termed the Ground Truth Sensor Model, employs as its input an array of objects delineated within the simulation frame, as furnished by the World Coordinate System (WCS). This model, representing the epitome of accuracy, utilizes the veritable values, dimensions, positions, velocities, orientations, and bounding boxes of the simulated entities.

The Phenomenological Sensor Model operates on principles similar to those of probabilistic models, while also integrating contextual effects. This approach offers a nuanced representation of real sensor dynamics. The complexity of these models is increased by the need to incorporate special phenomena into the sensor framework, and to correlate these phenomena with context-sensitive data from the virtual environment [11].

Physical sensor models are based on physical aspects and can be numerically complex. Therefore, they may require significant computational power and may not have real-time capabilities. Subsequent models use rendering techniques provided by the simulation framework as input and generate output raw data in the form of point clouds, which contain distance, intensity and time stamps. Several rendering techniques can generate synthetic LiDAR sensor raw data, including ray tracing, ray casting, rasterization (Z-buffer), and light paths [12].

Virtual sensors have a wide range of applications in analog test platforms. Chen et al. [13] used an integrated simulation and testing platform for self-driving vehicles. Their platform offers the possibility to test real vehicles in a closed test area. Their approach is characterized by the fact that the sensor signals (GPS, IMU, LiDAR, and camera) are derived from high-precision virtual simulation scenarios and processed as real driving commands by the real control unit in the vehicle. Ying et al. [14] used an in-vehicle loop simulator and testbed to functionally validate self-driving cars Vehicle sensors (camera, LiDAR and RADAR) are stimulated by signals generated based on a virtual traffic scenario. Thus, this test environment enables repeatable and fully manageable test scenarios.

Both commercial and open-source simulation platforms provide virtual sensor models that manifest varying degrees of fidelity. Among these, CARLA [15] stands out as an open-source simulation framework offering a variety of sensor models. Another notable platform is CarMaker/TruckMaker, [16] developed by IPG Automotive, which features a specialized Simulink interface encompassing libraries for diverse sensors, including both realistic and ideal models of LiDAR and RADAR. Similarly, Vector's DYNA4 [17] provides an assortment of virtual sensor models catering to

 ISBN: 978-1-68558-175-6

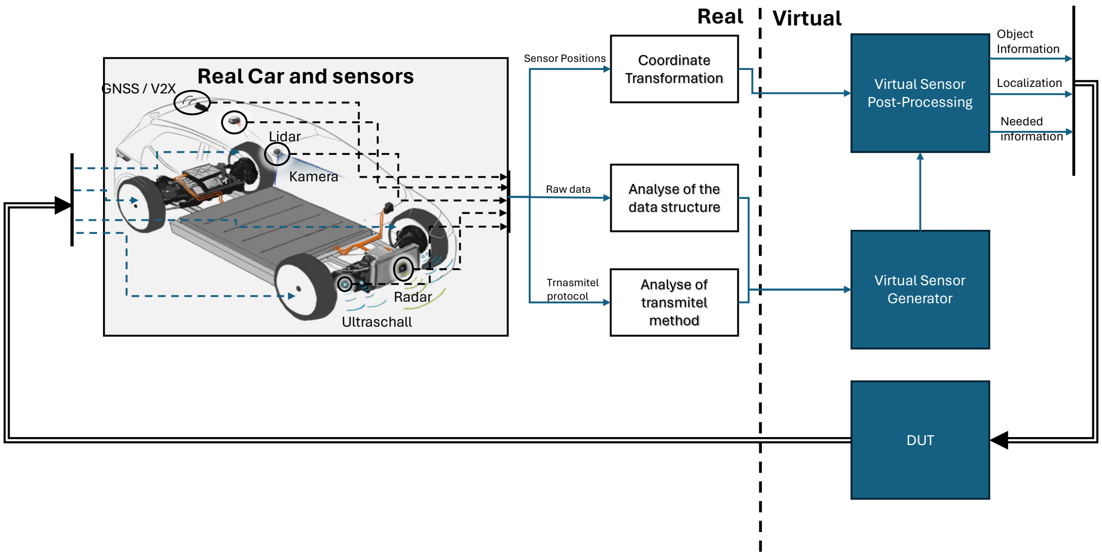

Figure 2. Concept of the sensor simulation and post-processing

LiDAR, ultrasound, and RADAR applications. Additionally, AURELION [18] by dSPACE extends its simulation capabilities by offering virtual models for LiDAR, RADAR, and camera sensors. Complementing these, the Automotive Simulation Models (ASM) models facilitate a broad spectrum of simulations, ranging from individual components like internal combustion engines or electric motors to comprehensive vehicle dynamics systems and intricate virtual traffic environments.

IV. CONCEPTION

This section presents the concept of virtual sensor simulation and post-processing. As shown in Figure 2, the system is divided into two parts: the real vehicle under test and the sensors that the vehicle has. The flowchart on the right outlines the process for simulating virtual sensors based on real sensor parameters. The process includes the following steps:

- **Analyze Data Structure**: Understand the organization, format, and internal relationships of the data.
- **Analyze Transmission Types**: Investigate how sensor data is sent, which may include communication protocols, data transfer rates, etc.
- **Coordinate Transformation**: This step converts the sensor data to a uniform coordinate system for comparison and analysis.
- **Virtual Sensor Generator**: This function block suggests a system or software to create a virtual model or representation of sensor data for manipulation or testing in a simulation.
- **Virtual Sensor Post-Processing**: After generating the virtual sensor data, this step involves further refinement or processing, which includes, among other things, feature extraction, filtering, calibration, and so on.
- **DUT**: The final block labeled "DUT" stands for Device Under Test, indicating that the processed virtual sensor data will eventually be used for testing purposes, such as testing a car's autopilot function or sensor fusion algorithms.

Where the vehicle control signals fed back from the DUT will be sent to the actual vehicle's driver, creating a closed-loop test.

V. IMPLEMENTATION OF THE SERNSOR SIMULATION

In this section, sensor simulation as well as post-processing implementation will be carried out based on the proposed concepts.

A. Hardware introduction

This section describes the sensor data types of the real vehicle under test in terms of data transfer types and rates. Figure 3 shows the RCP function carrier AURONA. The vehicle is equipped with four direct drives and a break-by-wire system. All four wheels can be driven, braked and steered individually. GPS and LiDAR are used for position detection. Objects are detected via camera, LiDAR, ultrasound, and

Figure 3. RCP function carrier AURONA

RADAR. In this paper, we focus on the Camera, LiDAR and GNSS.

In this case, the data in LiDAR is set of laser point cloud, which can be denoted by S. Every laser point cloud $\underline{s_i}$ contains a distance value in x, y and z axis and the intense. The configurable parameters are scanning frequency f_L as well as the scanning angles θ_h and θ_v, where θ_h is the horizontal scanning angle and θ_v is the vertical angle field. Equally important is the angular resolution in the vertical $\Delta\theta_v$ and horizontal directions $\Delta\theta_h$. Using these parameters, which can be obtained from the datasheet, the P_L parameter set can be formed. The LiDAR data is transmitted via Ethernet and the actual sampling frequency is 10Hz.

The data type in the camera is a matrix $\underline{\underline{M}}$, where the size of the matrix is determined by the length l and width w of the image. For the sensor simulation, the required camera parameters are the FOV (Field of View) θ_C, the information of the lens (aperture $f_{CL}$and focal length FL_{CL}) and the frame rate f_C. All parameters of the camera can be formed as a parameter set P_C. The transmission type is Ethernet, and the frequency is 20 FPS.

The IMU and GPS sensors are integrated in the GNSS, which provide the vehicle's dynamic state$\underline{x_V}$, and absolute coordinates in the geographic coordinate system$\underline{P_V}$. The vector $\underline{x_V}$ contains the acceleration of the vehicle in the Cartesian coordinate system along the axial direction $(a_{v_x}, a_{v_y}, a_{v_z})$ and the angular velocity as it rotates around the axis of the same Cartesian coordinate system $(\omega_{v_x}, \omega_{v_y}, \omega_{v_z})$, while the GPS provides the latitude lat_v, longitude $long_v$ and altitude h_v. All parameters of the camera can be formed as a parameter set P_{GNSS}. The transmission type is CAN-BUS and the measurement frequency can reach 100Hz.

In following simulations, it's needed to strive for the virtual sensor data to be consistent with the parameters of the real sensor data.

B. Coordinate system

In autonomous driving, the key to precise perception of the environment by the vehicle is that different sensor data can be expressed in a uniform coordinate system, so it is essential to define a uniform vehicle coordinate system and to find the position of the sensors in that vehicle coordinate. Therefore, we need to define the coordinates of the sensors as BCS_{Sensor}, and the coordinate system of the vehicle as BCS_v. The BCS_X coordinate system represents the body coordinate system. The transformation of the sensor coordinate system to the vehicle coordinate system can be expressed using the $\underline{\underline{T}}^V_{Sensor}$ matrix. The data $\underline{d_i}^{Sensor}$ in the sensor can be converted to the vehicle coordinate system by the (1).

$$\underline{d_i}^V = \underline{\underline{T}}^V_{Sensor} * \underline{d_i}^{Sensor} \quad (1)$$

C. Virtual Sensor Generator

Based on the previous state of arts, this paper describes the generation of virtual sensors using the advanced capabilities of the ASM model and AURELION. AURELION is a versatile software designed for simulating and visualizing sensor data. It facilitates the integration of actual sensor readings into various stages of development, testing, and validation processes for perception algorithms and driving functionalities. AURELION facilitates multiple development stages, such as hardware-in-the-loop (HIL) and software-in-the-loop (SIL), by providing flexible data interfaces that allow for the customization of virtual sensor parameters. Additionally, AURELION's open interface enables the retrieval and analysis of data from virtual sensors.

On the other hand, ASM offers a wide range of simulation models designed for automotive applications, which can be selectively integrated to meet specific requirements. ASM provides detailed insights into the vehicle's motion and displacement, allowing for the simulation of virtual IMU and GPS sensors. This approach, which integrates ASM and AURELION, provides a strong framework for accurately representing and analyzing vehicular dynamics and sensor systems.

D. Post -Processing

Different sensors require different post-processing methods. For camera sensors and LiDAR sensors the post-processing is feature extraction. In this work, the camera images are used to identify other traffic participants and traffic signals in the virtual environment based on the YOLO [18] algorithm. YOLO V8 is used in this article and notable for its speed and efficiency, dividing the image into a grid and simultaneously predicting bounding boxes and probabilities for each grid cell.

A key formula in YOLO calculates the confidence score for each bounding box, indicating the likelihood of object presence and the accuracy of the box location, which can be represented by the (2).

$$Conf(Obj_C) = P(Obj_c) \cdot IOU^{truth}_{pred} \quad (2)$$

Here, $P(Obj_c)$is the probability that an object exists within the box, and IOU^{truth}_{pred}represents the intersection over union between the predicted and the actual bounding boxes.

The virtual LiDAR sensor can obtain the object of the surrounding traffic participants and their location information in real time through Exwayz's [19] object recognition algorithm.

In virtual simulation, we cannot directly obtain the global coordinates under the Geographic coordinate system provided by GPS but are based on the global coordinates of the simulation environment $\underline{P_t^{VE}}$. To obtain GPS information, it is necessary to convert the coordinates in the simulation environment into GPS coordinates. Since the virtual simulation environment is built based on the real scene, the initial position of the Ego vehicle in the virtual environment can be obtained as $\underline{P_0^{VE}} = (x_0^{VE}, y_0^{VE})$, which corresponds to the GPS data $\underline{W_0} = (lat_0, long_0)$ in reality.

When the vehicle starts to move, the position of the vehicle at any moment in the virtual environment can be expressed by $\underline{P_t^{VE}} = (x_t^{VE}, y_t^{VE})$. The final GPS coordinate $\underline{W_t} = (lat_t, long_t)$ can be represented by (3) and (4).

 ISBN: 978-1-68558-175-6

$$lat_t = \frac{(y_t^{VE} - y_0^{VE})}{er} \cdot \frac{180}{\pi} + lat_0 \qquad (3)$$

$$long_t = \frac{(x_t^{VE} - x_0^{VE})}{er \cdot \cos\left(lat_0 * \frac{180}{\pi}\right)} \cdot \frac{180}{\pi} + long_0 \qquad (4)$$

Since the motion parameters of the car obtained from the simulation model are ground truth, which has no noise, while the data in the real IMU sensor includes zero-bias and random walk noise, the post-processing for the IMU sensor is to assign the motion state values to the real noise, whose noise values can be found in the manufacturer's parameter descriptions. Therefore, the simulation of real IMU values should be done by (5) and (6), where $\underline{\tilde{\omega}}$ and $\underline{\tilde{a}}$ are the noisy IMU measurement, b_g, b_a are the zero bias of the gyroscope and accelerometer, and n_g, n_a are their random walk noise.

$$\underline{\tilde{\omega}} = \underline{\omega} + b_g + n_g \qquad (5)$$

$$\underline{\tilde{a}} = \underline{a} + b_a + n_a \qquad (6)$$

E. *Result*

In this section, the results of the post-processing will be shown, as can be seen in Figure 4, where the vehicles and traffic signals at the crossroads in the image captured by the virtual camera are successfully detected, and the detected objects are boxed by the rectangular frame. After the object is recognized, its corresponding weight is displayed.

Figure 4. Object Detection in the camera image

The recognition of objects in the virtual LiDAR point cloud is illustrated in Figure 5, where the recognized objects are boxed by cubes.

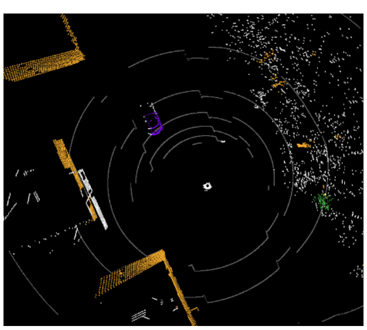
Figure 5. Object Detection in the Point-Cloud

Figure 6 and Figure 7 show the data from the virtual IMU. In order to make the noise in the IMU data more visible, a release frequency of 100 Hz (the same as the real device) was used to simulate two hours of IMU stationary. In that case, the measurements of the noiseless IMU should be 0 except for the z-axis acceleration, which receives the effect of gravitational acceleration. With the added noise, the value of this IMU is around 0 and z -axis acceleration is around the -9.81.

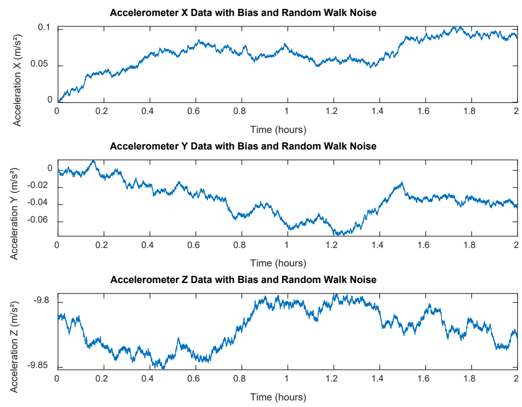

Figure 6. Noisy IMU data-acceleration

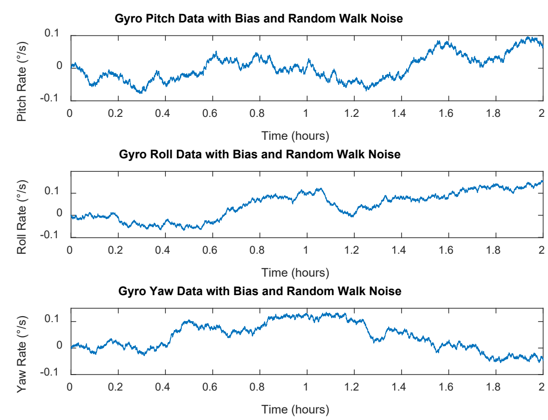

Figure 7. Noisy IMU data - angular velocity

Figure 8 shows the GPS values. Loading the converted GPS path into OpenStreetMap shows that its virtual GPS data basically matches the real driving path.

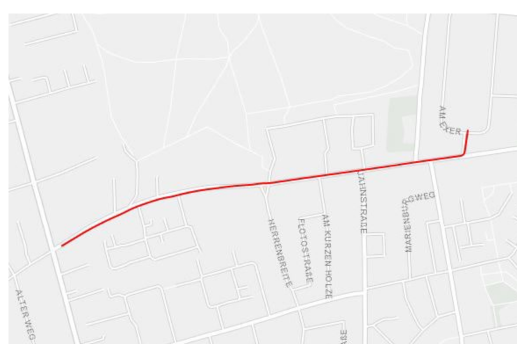
Figure 8. Simulated GPS path on the map

VI. CONCLUSION AND FUTURE WORK

This paper describes the simulation of virtual sensors in ERAGON, a highly configurable real-time test system for intelligent vehicles that simulates autonomous driving environments together with the functional vehicle AURONA. The paper focuses on camera, LiDAR, GPS and IMU sensors and post-processing techniques including feature extraction and noise modeling to refine the virtual sensor data for practical applications. The results section illustrates the effectiveness of the system in detecting vehicles and traffic signals through virtual sensor data, demonstrating the potential of virtual simulation in enhancing the design and testing of self-driving car technologies. The following work will continue to refine the techniques for virtual sensor simulation and post-processing, and fusion of multiple virtual sensors.

ACKNOWLEDGMENT

Funded by the Lower Saxony Ministry of Science and Culture under grant number ZN4172 in the Lower Saxony Advance of the Volkswagen Foundation.

VolkswagenStiftung

Niedersächsisches Ministerium
für Wissenschaft und Kultur

REFERENCES

[1] X. Liu-Henke, O. A. Yarom, and S. Scherler, “Virtual Development and Validation of a Function for an Automated Lateral Control using Artificial Neural Networks and Genetic Algorithms,” in 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 2020, pp. 1–5

[2] X. Liu-Henke, M. Göllner, S. Jacobitz, and J. Zhang, “Konzept eines ganzheitlichen, hochflexibel konfigurierbaren HiL-Prüfsystems für den Test autonomer Fahrfunktionen”, 01 2023, pp. 71–76.

[3] A. Elmquist and D. Negrut, “Methods and Models for Simulating Autonomous Vehicle Sensors,” in IEEE Transactions on Intelligent Vehicles, vol. 5, no. 4, pp. 684-692, Dec. 2020, doi: 10.1109/TIV.2020.3003524.

[4] X. Liu-Henke, S. Scherler, and M. Goellner, “System architecture of a full active autonomous electric vehicle,” in 2017 IEEE Transportation Electrification Conference and Expo (ITEC), 2017, pp. 51–56.

[5] X. Liu-Henke, S. Jacobitz, M. Göllner, J. Zhang, S. Scherler, and O. A. Yarom, “Cyber-physical Industry 4.0 laboratory test field to simulate self-optimizing intralogistics,” in 2020 19th International Conference on Mechatronics - Mechatronika (ME), 2020, pp. 1–6.

[6] S. Jacobitz, M. Gollner, J. Zhang, O. A. Yarom, and X. Liu-Henke, “Seamless validation of cyber-physical systems under real-time conditions by using a cyber-physical laboratory test field,” in 2021 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE), Shanghai, China, 2021, pp. 1–8.

[7] S. Jacobitz and X. Liu-Henke, “A Real-Time Interface for Xcos – Demonstrated on a Battery-management System,” in 2nd Scilab Conference, Berlin, Germany, 2019.

[8] M. Aeberhard et al., ‘Experience, Results and Lessons Learned from Automated Driving on Germany’s Highways’, IEEE Intelligent Transportation Systems Magazine, vol. 7, no. 1, pp. 42–57, 2015.

[9] S. Mitsch, K. Ghorbal, and A. Platzer, “On provably safe obstacle avoidance for autonomous robotic ground vehicles,” in Robotics: Science and Systems IX, 2013.

[10] W. Wachenfeld and H. Winner, “The Release of Autonomous Vehicles,” in Autonomous Driving, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 425–449.

[11] M. Hartstern, V. Rack, M. Kaboli, and W. Stork, “Simulation-based Evaluation of Automotive Sensor Setups for Environmental Perception in Early Development Stages,” 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 2020, pp. 858-864.

[12] A. Haider et al., “Development of high-fidelity automotive LiDAR sensor model with standardized interfaces,” Sensors (Basel), vol. 22, no. 19, p. 7556, Oct. 2022.

[13] S. Chen, Y. Chen, S. Zhang, and N. Zheng, “A Novel Integrated Simulation and Testing Platform for Self-Driving Cars With Hardware in the Loop,” IEEE Trans. Intell. Veh., vol. 4, no. 3, pp. 425–436, 2019.

[14] Y. Gao, X. Zhao, Z. Xu, J. Cheng, and W. Wang, “An Indoor Vehicle-in-the-Loop Simulation Platform Testing Method for Autonomous Emergency Braking,” Journal of Advanced Transportation, vol. 2021, pp. 1–12, 2021.

[15] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An Open Urban Driving Simulator,” in Proceedings of the 1st Annual Conference on Robot Learning, 13--15 Nov 2017, vol. 78, pp. 1–16.

[16] D. Deter, C. Wang, A. Cook, and N. K. Perry, “Simulating the Autonomous Future: A Look at Virtual Vehicle Environments and How to Validate Simulation Using Public Data Sets,” in IEEE Signal Processing Magazine, vol. 38, no. 1, pp. 111-121, Jan. 2021.

[17] Vector, “Sensor Simulation for Virtual Test Driving,” https://j2n.eu/r/CCpvn. Accessed 16 Mar. 2024.

[18] dSPACE, “AURELION - Sensor-Realistic Simulation,” dSPACE, https://j2n.eu/r/W3C5A. Accessed 16 Mar. 2024.

[19] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788.

[20] Exwayz, “Exwayz 3D mapping: software for creating dense, accurate and georeferenced 3D point cloud using LiDAR SLAM,” 2024. [Online]. Available: https://www.exwayz.fr/. Accessed 16 Mar. 2024.