

---

# Privacy and Performance in Virtual Reality: The Advantages of Federated Learning in Collaborative Environments\*

---

Daniel Flores-Martin<sup>1,\*</sup>, Francisco Díaz-Barrancas<sup>2</sup>,  
Pedro J. Pardo<sup>2</sup>, Javier Berrocal<sup>2</sup> and Juan M. Murillo<sup>2</sup>

<sup>1</sup>*COMPUTAEX. Extremadura Supercomputing Center, Cáceres, Spain*

<sup>2</sup>*University of Extremadura, Badajoz, Spain*

*E-mail: daniel.flores@computaex.es; frdiaz@unex.es; pjparado@unex.es;  
jberolm@unex.es; juanmamu@unex.es*

*\*Corresponding Author*

Received 27 September 2024; Accepted 09 November 2024

## Abstract

Federated Learning has emerged as a promising approach for maintaining data privacy across distributed environments, enabling training on a diverse range of devices from high-performance servers to low-power gadgets. Despite its potential, managing numerous data sources can strain these devices, particularly those with limited capabilities, leading to increased latency. This is especially critical in virtual reality, where real-time responsiveness is crucial due to the need for constant data connectivity. Historically, virtual reality systems have relied on tethered computer setups, restricting their flexibility and the benefits of wireless technology. However, recent

---

\*This work is an extension of a previous paper: Flores-Martin, D.; Díaz-Barrancas, F.; Pardo, Pedro J.; Berrocal, J.; Murillo, J.M. Federated Learning on Virtual Reality Environments: Performance Analysis on Standalone Devices. The 4th International Workshop on Big data driven Edge Cloud Services (BECS 2024) Co-located with the 24th International Conference on Web Engineering (ICWE 2024), June 17-20, 2024, Tampere, Finland. <https://link.springer.com/book/9783031751097>

*Journal of Web Engineering, Vol. 23\_8, 1085–1106.*

doi: 10.13052/jwe1540-9589.2382

© 2025 River Publishers

advancements have enhanced the computational power of VR devices, allowing them to perform certain tasks independently. This work explores the feasibility of training a neural network on VR devices, using a federated learning approach, to develop a collaborative model aggregated and stored in the cloud. The goal is to assess the computational demands and explore the potential and constraints of leveraging VR devices for artificial intelligence applications.

**Keywords:** Virtual reality, federated learning, neural networks, training, cloud computing.

## 1 Introduction

Federated learning (FL) has become highly significant in tackling the dual challenge of utilizing distributed data to train machine learning (ML) models while ensuring data privacy. This allows devices to collaboratively train algorithms without sharing sensitive information centrally [21].

In an era where personal information is extremely valuable and privacy concerns are on the rise, FL stands as a fundamental pillar for advancing artificial intelligence (AI) and ML. This innovative approach not only drives the development of AI and ML models but also ensures the protection of sensitive data confidentiality. By allowing multiple devices to collaborate in training algorithms without the need to centrally share critical data, FL effectively addresses the tension between leveraging distributed data and preserving privacy. This method not only fosters greater technology innovation but also provides a robust solution to the challenges of securely managing personal information in an increasingly complex digital environment [19]. Numerous devices and technologies are available to exploit FL [2]. One such technology is the virtual reality (VR). The growing popularity of VR stems from its ability to customize immersive environments, providing users with highly individualized and engaging experiences [29]. Also, VR enables users to access and enjoy immersive experiences without broadly sharing personal data, positioning it as a robust tool for protecting privacy. Moreover, this technology can be integrated into various fields such as healthcare [22], education [12], or security [23].

In this context, the convergence of FL and VR has created a significant paradigm shift in the development and enhancement of ML models. This intersection allows users to experience highly personalized services and scenarios while maintaining robust privacy protection. By leveraging FL, VR

systems can train models on user data locally across multiple devices, ensuring that sensitive information is never shared or centralized. This approach not only enhances the accuracy and relevance of virtual experiences but also ensures that users' personal data remains secure. Furthermore, this synergy opens up new possibilities for creating immersive and customized applications across various fields, from entertainment to healthcare, all while upholding stringent privacy standards [33].

FL in VR enables decentralized model training on users' devices, with cloud aggregation for a global model. This approach preserves privacy, reduces latency, and enhances efficiency. Optimizing FL for VR could expand its use in education, medicine, and industrial training, where personalization and privacy are crucial. In addition, the immersion offered by VR enables richer data capture and enables real-time collaboration in a shared virtual environment. Thus, users can contribute to model improvement by interacting in the VR environment and generating data in real-time, which improves the quality and diversity of the data used for FL. The cloud, on the other hand, plays a key role in grouping the models generated in the VR devices into a global model. Training FL models directly on VR devices is interesting for privacy, as sensitive data remains on each device. This decentralized approach reduces bandwidth, boosts efficiency, and allows user-specific model personalization while enhancing a global model. FL is scalable, distributing processing across devices and optimizing resources.

This paper addresses the potential of training FL models directly on autonomous VR devices, utilizing data collected from users to develop personalized models. By decentralizing the training process to local devices, the approach reduces the strain on centralized servers and overcomes connectivity constraints, facilitating enhanced collaboration and cooperative learning. The key contributions of this work are:

- **Neural network design:** Proposing a custom neural network (NN) focused on classification, serving as an initial prototype for device testing.
- **Training assessment:** Evaluating training performed on VR devices with varying characteristics to observe and analyze device behavior.
- **Resource analysis:** Measuring resource usage across energy consumption, CPU load, and network utilization.

The remainder of the paper is organized as follows: Section 2 introduces the background and key motivations behind the work. Section 3 describes the proposed federated virtual reality environment. In Section 4, the detailed

results of the work are presented. Section 5 reviews relevant related work. Finally, Section 6 provides a summary of the conclusions and final reflections.

## 2 Background and Motivations

Training ML algorithms and neural networks (NNs) on standalone VR devices have garnered significant attention due to its rise, importance, and multiple benefits [11]. For instance, a VR system with deep learning (DL) can now adjust lighting, sound, and other environmental aspects to suit user preferences, creating a more immersive experience. Advances in device processing power allow complex tasks to be handled locally, enabling direct ML model deployment and ushering in a new era of personalized VR applications.

Training neural networks on standalone VR devices offers several benefits. It enhances user experience through personalized, adaptive interactions [25], and improves immersion by adjusting to user behavior in real-time [18]. Local training reduces network and server costs while optimizing device resources [1]. It also enables innovative applications, such as adaptive educational tools [17], and supports continuous model updates directly on the device, improving functionality over time.

On the one hand, VR technology has advanced from recreational use to professional applications [9, 31]. Integrating NNs enhances real-time adaptability in VR. Our focus is on standalone devices, utilizing their hardware to run NN algorithms directly within the headset, removing the need for external computing resources.

On the other hand, FL is closely linked to distributed learning where scaling can be vertical and horizontal [2] promoting proactive smart environments [28]. The initial FL proposal, aiming to enhance the model for Android clients, resembles distributed computing. While FL prioritizes safeguarding privacy, contemporary research in distributed ML also underscores the development of privacy-preserving distributed systems [20]. Distributed processing involves connecting several computers in different locations through a communication network controlled by a central server. Each computer performs different parts of the same task to complete it. Therefore, distributed processing seeks to speed up the processing stage, while FL focuses on building a collaborative model without privacy leaks, even in low-power devices or devices with low computational capabilities [16]. FL can be applied in different domains such as education, shopping, healthcare, industry, etc. [32]. This situation poses two key challenges. First, data barriers between different entities make it difficult to directly aggregate data for model

training, ensuring data privacy and security. Second, the heterogeneity of data stored by these parties makes it difficult for traditional ML models to operate efficiently. Therefore, FL emerges as an essential solution to these problems, by leveraging the features of FL, an ML model can be built for the parties involved without exporting enterprise data, ensuring both data privacy and security while delivering personalized services.

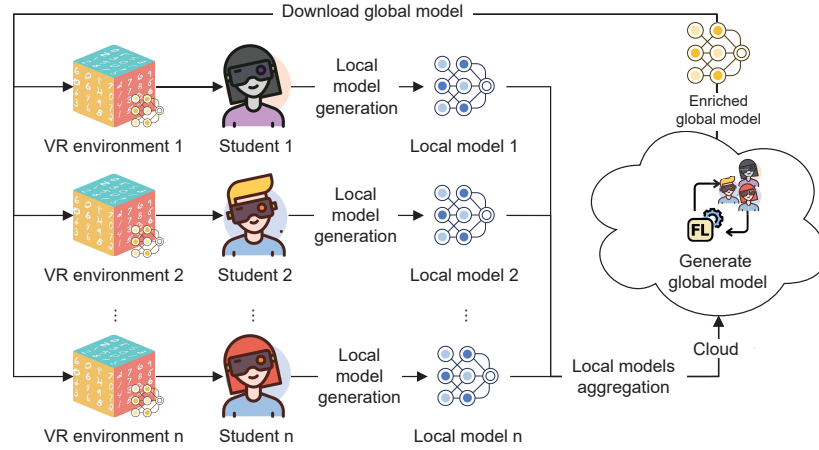
Therefore, combining VR with FL is an interesting option when training advanced NNs distributively and protecting users' privacy.

## 2.1 Use Case

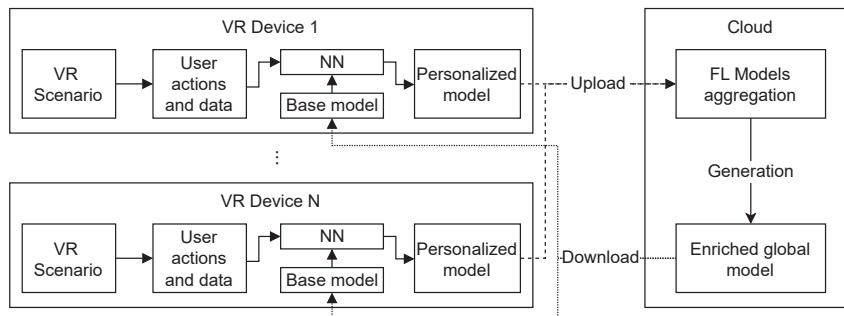
To show this, a use case based on an educational environment is proposed, where students use VR and FL to perfect the writing of handwritten digits.

*Let's imagine an educational environment, for example, in a mathematics class or as part of a rehabilitation program for people with motor difficulties, where students learn to write digits in VR, continuously improving recognition accuracy as they interact with the system. In this setting, students practice writing digits (from 0 to 9) using VR devices that capture their movements in the air. Initially, each student trains a local digit recognition model based on a handwritten numbers dataset, adapted to their writing style, as writing in VR can vary in size, speed, and stroke. Later, FL comes into play, allowing each user's device to send their locally adjusted models to a central server, which combines these updates into a global model. This model, enriched with information from the variations in each user's writing style, is redistributed to all devices, benefiting everyone with a more accurate and robust model. This continuous improvement ensures that initial users and new students experience enhanced learning with a more precise and efficient recognition system. Additionally, this system provides personalized feedback to each student, allowing them to correct errors in digit formation and helping them improve over time. Meanwhile, teachers can monitor student progress and offer support in specific areas. FL ensures that sensitive user data, such as their writing style, remains private, while the global model is constantly enriched through collaboration from all participants.*

Figure 1 illustrates an educational environment in VR where users learn to write digits in the air using VR devices. Initially, each user trains a local digit recognition model based on the MNIST dataset, adapted to their writing style. Then, the local models are sent to a central server that combines them into a global model, benefiting all users with a more accurate recognition system. This allows for continuous improvement while preserving data privacy, and is applicable in both educational contexts and rehabilitation programs.



**Figure 1** Use case details: Users practice writing digits with VR devices and generate their FL models aggregated in the cloud.



**Figure 2** Approach diagram for VR and FL.

### 3 Federated Virtual Reality Environment

The proposed approach is illustrated in Figure 2. On the one hand, VR devices, through user actions and data, generate their customized model taking a base model as a reference, which in the first instance will be a simple model that will evolve into a more enriched version. On the other hand, the cloud receives these customized models to unify them and generate an enriched global model, which will be made available to VR devices to serve as a base model for training.

For better understanding, this section is structured to provide an in-depth understanding of our VR scenario and the training process. To this end, the NN employed, the VR scenario designed, and the tests are detailed below.

### **3.1 NN Architecture**

A feed-forward NN was developed using *C#* and the Unity engine. Since no existing libraries support real-time training in VR, we built our NN directly within a virtual scene development platform, eliminating the need for pre-trained networks. The Modified National Institute of Standards and Technology (MNIST) [10] dataset, which is widely used in machine learning research and consists of handwritten digit images, was chosen for this project. This dataset and NN were selected to serve as a proof of concept for training networks on VR devices and to evaluate their performance. The ultimate goal is to refine this approach with real users in a virtual educational center. Given that this is a simple NN, the chosen dataset aligns well with the network's characteristics. Future iterations are expected to incorporate more complex parameters, allowing the network to adapt more realistically to the presented use case.

The NN consists of a multi-layer structure with two hidden layers, each containing 150 neurons, an initial layer with 784 neurons, and an output layer with 10 outputs corresponding to digits from 0 to 9. To optimize the performance of our NN, we engaged in 10 epochs of training, a learning rate set at 0.1, and a batch size of 8. These parameters were chosen to strike a delicate balance between the network's ability to learn from the training data, converge efficiently, and generalize well to unseen data. The epoch count reflects the number of times the entire training dataset is passed through the network, the learning rate that governs the size of the steps taken during the optimization process, and, finally, the batch size representing the number of training samples utilized in each iteration.

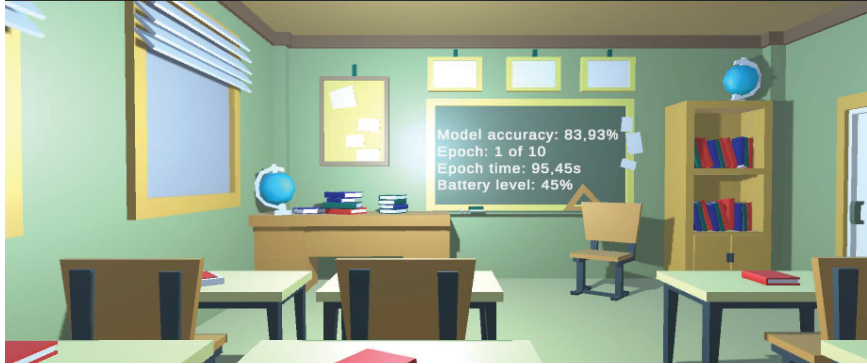
Implementing a neural network for handwritten digit prediction in a VR and FL study highlights its potential for personalized learning and rehabilitation. It evaluates FL's performance in VR, focusing on accuracy and real-time efficiency, while simulating user interaction. This setup demonstrates the feasibility of decentralized training and explores scalability in dynamic applications.

### **3.2 VR Scene and Training Procedure**

A classroom<sup>1</sup> was adapted to simulate a VR scenario in which different objects can be rendered while NN training occurs in the background (Figure 3).




---

<sup>1</sup><https://sketchfab.com/3d-models/class-room-01-82758fa4a5864744833e01c723a1cf4e>



**Figure 3** A VR scene where an NN model is trained with the MNIST database of handwritten numbers. Different parameters such as model accuracy, epoch number, time of the epoch, and the battery of the device are displayed.

**Table 1** Technical specifications of the different VR devices used

			
	Meta Quest	Meta Quest 2	Meta Quest 3
Release Date	May 2019	October 2020	October 2023
System on a chip	Snapdragon 835	Snapdragon XR2	Snapdragon XR2 2nd Gen
RAM memory	4GB	6GB	8GB
Graphics (GFLOPS)	Adreno 540 (545-567)	Adreno 650 (up to 1320)	Adreno 740 (up to 1740)
Firmware version	v49	v54	v59

To train the network, the database digits were downloaded for model training and testing. Training and accuracy measurement occur in real-time during virtual scenario rendering. Upon completion, network data is saved in JSON format, allowing pre-trained networks to be loaded and further refined on other devices. VR software uses customer feedback processed within headsets and syncs to the cloud, supporting a FL model that leverages insights from multiple users.

Different VR devices were selected to test the proposed approach. Three devices were chosen from the same brand but of different generations: Meta Quest, Meta Quest 2, and Meta Quest 3 (Table 1). The processor, the RAM, the graphics, and the firmware version were considered for these devices. The NN configurations for the detailed scenario and devices are shown below.



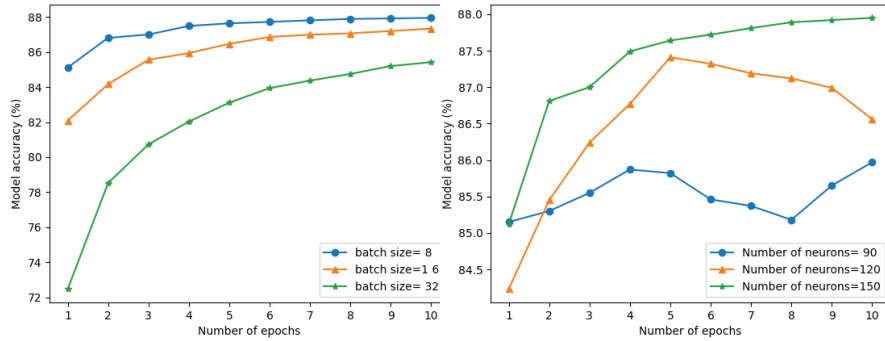
### 3.3 Tested NN Settings

In this section, we describe the methodology that guided our decision-making process when selecting the NN architecture for our system. The primary goal of our approach was to maximize accuracy, which was critical given the system's complexity and its application in VR. We outline a methodology based on the following key steps and considerations:

1. **Define the primary objective:** This approach prioritized maximizing neural network accuracy. While factors like energy efficiency and processing time were considered, they were secondary in this context.
2. **Evaluate the trade-offs:** Trade-offs in the NN architecture were evaluated, focusing on accuracy versus battery drain and execution time. A more complex network improves accuracy but increases power consumption, which we deprioritized due to low energy concerns. Although larger networks slow down execution, our VR system's real-time constraints allowed us to prioritize accuracy over speed.
3. **Simplify the architecture for practicality:** While aiming for high accuracy, we balanced ease of implementation by choosing a simpler neural network structure. This reduced complexity, enabling easier debugging, testing, and faster iteration without sacrificing accuracy.
4. **Establish parameters:** Key parameters were defined to guide architecture selection: the number of layers, kept within a range to ensure capacity while avoiding overfitting; the number of neurons per layer, determining computational capacity; and the batch size, which controlled how many samples were processed before updating the model's weights.
5. **Fix settings rules:** Different rules were considered to adjust the parameters of the network in order of priority: do not overflow the device's memory, to avoid crashes or blocks in the execution of the application, and to prioritize the highest possible accuracy above execution time.

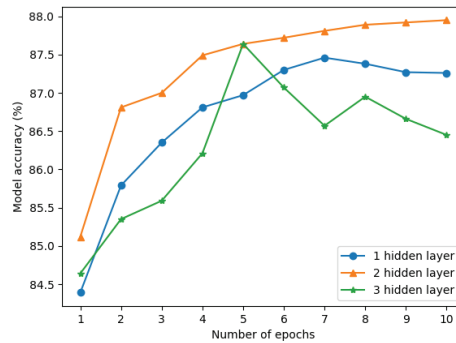
Once the methodology was established, several tests were conducted that allowed us to determine the best configuration for the chosen NN architecture.

Figure 4a shows the network's convergence towards optimal accuracy. Batch size, determining the number of samples propagated through the NN, affects memory usage and accuracy inversely. We set the batch size to 8 for this analysis. Next, we determined the number of neurons in hidden layers, impacting network complexity, as shown in Figure 4b. Fewer neurons prompt faster convergence while increasing neuron count delay convergence. After observing continued improvement beyond 10 epochs, we settled on 150



(a) Batch sizes.

(b) Neurons for batch size of 8.



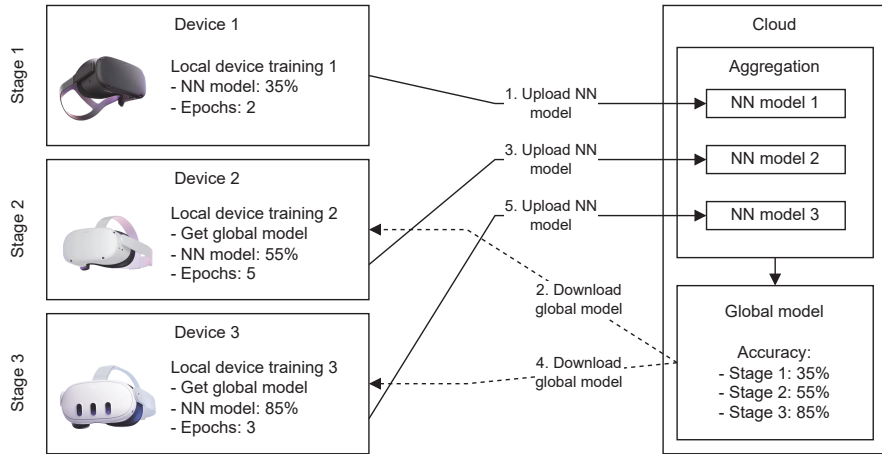
(c) Hidden layers for batch size of 8 and an 150 neurons.

**Figure 4** Evaluation of different metrics.

neurons in hidden layers (Figure 4c). Finally, the number of hidden layers also influenced complexity. The configuration with two layers yielded the best results. Also, the same number of epochs and learning rate were set for all configurations.

### 3.4 Cloud as a Bridge Towards FL Using VR

In our work, the cloud plays a crucial role within the FL system. VR devices, despite their limited computational resources and the challenge of conserving battery life, can train customized models locally without significant battery depletion. These locally trained models are then uploaded to the cloud, where they are aggregated to produce a more refined global model. This approach



**Figure 5** Schematic of our FL proposal in VR using the cloud as a binding.

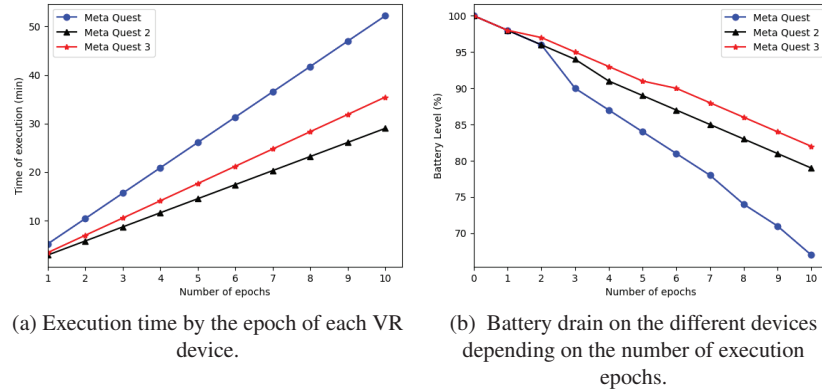
results in a global model with enhanced accuracy, benefiting from the diverse data contributions of individual devices.

Figure 5 shows an interpretation of aggregating the FL model using VR and load sharing by updating the trained model in the cloud. In this scenario, we have three independent VR devices. According to user data, each device can train the model for a defined number of epochs. The results will be detailed in the next section.

## 4 Results

Execution time and battery consumption are selected to evaluate the proposal’s performance. On the one hand, Figure 6a represents the execution time for the different VR devices. It shows a significant evolution between the Meta Quest device and Meta Quest 2 (around 45%). However, it can be observed that the new Meta Quest 3, which is promised to be 2.5 times more powerful than the one in Meta Quest 2 due to the second generation of CPU and new GPU,<sup>2</sup> gets worse execution times (around 18%). The explanation provided by Meta, which seems more coherent, is that with the device’s firmware update, they will be deploying their full potential as they did with their predecessors that were previously less powerful.

<sup>2</sup><https://blog.learnxr.io/extended-reality/quest-3-review-and-developer-setup>

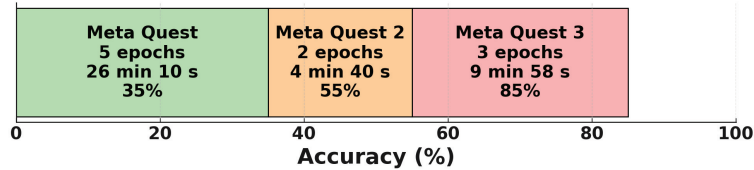


**Figure 6** Analyzed metrics.

On the other hand, Figure 6b shows the evolution of the battery drain of each VR device over the number of epochs (taking into account the battery usage involved in rendering and interacting with the scene). In this case, the last-generation Meta Quest 3 performs better than its predecessors, Meta Quest 2 and Meta Quest. The reduction in battery consumption between the Meta Quest headset and Meta Quest 3 is more than 15%. If we analyze the difference in battery consumption between Meta Quest 3 and its predecessor Meta Quest 2, the reduction is about 4%. In addition, we analyzed the case where the scenario was reproduced without NN training to compare NN's influence on battery drain directly. For all cases, the task of training the NN was 2% compared to the same scene without training, so performing the training does not incur a high battery drain cost.

VR technology is subject to continuous updates, both in terms of hardware enhancements and firmware optimizations. This fluidity introduces unpredictability into the user experience, as the true potential of a device may only be realized through iterative updates and improvements.

In the experiment detailed in Figure 5, we obtained varying accuracy levels. Specifically, Meta Quest achieved an accuracy of 35% after 2 epochs with a training time of 4 minutes and 40 seconds. Subsequently, using Meta Quest 2, the model was further trained for 5 epochs, reaching an improved accuracy of 55% in 26 minutes and 10 seconds. Finally, Meta Quest 3 performed the final retraining with 3 epochs, culminating in an accuracy of 85% within 9 minutes and 58 seconds. These findings highlight the impact of device performance, training epochs, and time allocation on model accuracy (Figure 7).



**Figure 7** Accuracy achieved at different training stages using three devices (Meta Quest, Meta Quest 2, and Meta Quest 3) based on the number of epochs and training time.

## 5 Related Work

The application of FL in VR poses several complex challenges, such as the need to train models with non-identically distributed data across different devices, the requirement for efficient resource allocation, and platform compatibility. To address these challenges, recent works have delved into integrating the FL framework for resource allocation within the metaverse [36] and leveraging FL for real-time object detection in an edge computing environment [6].

A review of the current state of FL in VR is provided in [8]. This study focuses on resource allocation for personalized VR services tailored to users with varying needs within a mobile edge computing system. The authors introduce a quality of experience (QoE) metric that accounts for factors such as latency, user attention levels, and preferred resolutions.

Moreover, NN in VR devices has been previously explored with pre-trained algorithms, where the network is solely employed for real-time behavior prediction in VR scenarios [30]. Additionally, works are focused on advancing DL models by proposing innovative methods that explore and implement visualization technologies at the intersection of DL and VR research projects [26]. An interesting initiative is the application of federated DL to predict user locations and orientations in VR, aiming to minimize VR user load and improve overall user experience [7]. Various use cases have been created in this field to simulate real-life situations. For example, they use DL and VR to address the attentional patterns of English learners to establish a correlation between the linear control model and their task performance [35].

Previous works have demonstrated the value of combining VR and FL, acknowledging the benefits both fields bring when integrated. However, a key limitation of these studies is that they do not propose performing model training directly on VR devices in a federated manner. This represents a significant distinction from our approach, as we focus on enabling distributed,

on-device training in VR environments. By doing so, we can enhance data privacy, reduce network dependency, and improve model personalization at the device level – critical factors in immersive, real-time VR applications. The convergence of VR and FL thus opens up new possibilities for developing more scalable and secure solutions, where sensitive user data never leaves the device, and models are continuously improved through decentralized collaboration. This unique intersection of VR and FL offers fertile ground for further research, making it an intriguing and underexplored opportunity in areas like healthcare, education, entertainment, and remote collaboration [5, 24, 34].

Also, analyzing the FL techniques that are being used in combination with VR we have the following use cases:

- **Real-time object detection:** In edge and cloud computing environments, FL is leveraged for real-time object detection in VR, demonstrating practical applications of FL in dynamic scenarios.
- **Behavior prediction:** VR devices also utilize pre-trained NNs for real-time behavior prediction, while AR applications extend this to user experience enhancements through adaptive FL algorithms.
- **User location and orientation prediction:** Some studies focus on DL to predict user locations and orientations in VR, aiming to reduce user load and enhance the overall experience.

Table 2 highlights some significant works related to our research and will showcase the main differences and similarities with this work. The analyzed features provide a comprehensive overview of the capabilities and characteristics of different VR devices using neural networks. It analyzes whether the devices are standalone or rely on external hardware, which affects their autonomy and ease of use, and it examines whether inference (making predictions with a trained model) is performed directly on the device, which is crucial for latency and real-time processing capabilities. Also, the table details the type of predictions made, and it evaluates whether the neural network is scalable, meaning it can handle an increase in model size or complexity without compromising performance. Finally, it considers whether the processing is performed in real-time, and it indicates whether any part of the processing is done in the cloud, which may imply greater computing power but also depends on a robust and reliable network connection.

Our experiments focused on training a federated NN directly on VR devices. As a complement, we established two performance metrics: runtime

**Table 2** Comparison table of related works

	Elor et al. [14]	Dohan et al. [13]	This work
<b>Device model</b>	HTC Vive Pro	HTC Vive Pro Eye	Meta Quest (1,2 and 3)
<b>Standalone</b>	No	No	Yes
<b>Inference</b>	Yes, using ML agents	No	Yes
<b>Training</b>	No	No	Yes
<b>Predictions</b>	Ghost arm	Real walking data	Writing of handwritten numbers
<b>Scalable NN</b>	Yes	Yes	No
<b>Real-time processing</b>	Yes	Yes	Yes
<b>Cloud processing</b>	No	No	Yes, to unify FL models

and battery consumption. These factors are essential for both user experience and the operational efficiency of devices. The results revealed that, despite limitations, training partial models on VR devices could be a promising strategy to enrich global models. In this sense, active user participation provides valuable information that enhances local models.

As model complexity increases, resource consumption rises, posing challenges for VR devices with limited processing power, memory, and energy. To address this, we began with a simple NN model. Though the approach was not without challenges, it allowed us to assess VR's response to NN training without interruptions due to excessive resource use. The simplified model demonstrated how NN-based applications can operate efficiently in VR environments with constrained resources, especially in low-consumption, real-time scenarios. It is important to highlight that the results cannot be directly compared with other works, as those typically train models externally. The key distinction in this approach is that the training occurs directly on the device, thus preserving privacy and eliminating the need to share data externally.

Although it is a valid first step in the effort to discover more, the approach has given evidence that more extensive strategies are necessary. The next scientific work should mainly focus on how to create models with the same high performance but lower resource usage. The tools such as model compression, pruning, or using lightweight architectures like mobile NNs in the form of exploration of this problem could be applied. Last but not least, learning that is adjusted so that NN complexity is dynamically tuned depending on resources available in VR devices can add to the above.

This illustrates that finding the proper balance between the complexity of NN models and resource constraints in VR environments is a very pressing one. The more VR systems develop, the greater is the need for more immersive experiences and scalable solutions that can incorporate advanced NNs while adhering to the device limitations that are essential for delivering seamless, high-quality VR applications.

## 6 Conclusions

This work demonstrated the feasibility of training AI algorithms on standalone VR devices through an FL system. After implementing an NN to run on the Unity graphics engine, we conducted extensive testing to confirm that the network operates in parallel with the main task without causing performance degradation or blocking other system functions. This stability is largely attributable to the inherent simplicity of the developed NN and the careful implementation of multithreading. The use of threads allows the main program to continue its execution flow without interruption, while efficiently utilizing the available processing capacity for parallel tasks. Additionally, we measured the execution time of the NN for the MNIST dataset on each device, and due to the importance of energy efficiency VR scenarios, we meticulously measured the battery consumption associated with the NN operations on each device. This aspect of our evaluation is critical, as it provides insights into the practicality of deploying such models in real-world applications where power constraints are a significant concern. By integrating both the computational performance and power efficiency metrics, we offer a comprehensive evaluation that can guide the selection of suitable hardware platforms for deploying neural networks in energy-sensitive environments.

In future work, we plan to create a VR scenario where users can interact with real clothing data, generating training data through interactions. In addition, new VR devices will be included in the work, and a specific cloud system for aggregating customized models will be developed.

## Acknowledgement

This work was supported by the projects PID2021-124054OB-C31, TED2021-130913B-I00, PDC2022-133465-I00 (MCIU/AEI/FEDER, UE), by the Department of Economy, Science and Digital Agenda of the Government of Extremadura (GR21133 and IB20094), by the European Regional



Development Fund, and by INCIBE and the “European Union NextGenerationEU/PRTR” (C110.23). All authors contributed equally.

## References

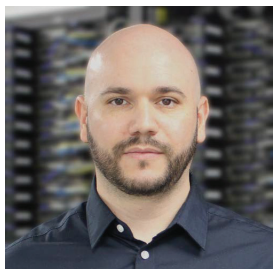
- [1] AbdulRahman, S., Tout, H., Ould-Slimane, H., Mourad, A., Talhi, C., Guizani, M.: A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet of Things Journal* **8**(7), 5476–5497 (2020)
- [2] Aledhari, M., Razzak, R., Parizi, R.M., Saeed, F.: Federated learning: A survey on enabling technologies, protocols, and applications. *IEEE Access* **8**, 140699–140725 (2020)
- [3] Angelov, V., Petkov, E., Shipkovenski, G., Kalushkov, T.: Modern virtual reality headsets. In: 2020 International congress on human-computer interaction, optimization and robotic applications (HORA). pp. 1–5. IEEE (2020)
- [4] Bejani, M.M., Ghatee, M.: A systematic review on overfitting control in shallow and deep neural networks. *Artificial Intelligence Review* **54**(8), 6391–6438 (2021)
- [5] Bhugaonkar, K., Bhugaonkar, R., Masne, N.: The trend of metaverse and augmented & virtual reality extending to the healthcare system. *Cureus* **14**(9) (2022)
- [6] Brecko, A., Kajati, E., Koziorek, J., Zolotova, I.: Federated learning for edge computing: A survey. *Applied Sciences* **12**(18) (2022), <https://www.mdpi.com/2076-3417/12/18/9124>
- [7] Chen, M., Semiari, O., Saad, W., Liu, X., Yin, C.: Federated deep learning for immersive virtual reality over wireless networks. In: 2019 IEEE Global Communications Conference (GLOBECOM). pp. 1–6 (2019).
- [8] Chen, Y., Huang, S., Gan, W., Huang, G., Wu, Y.: Federated learning for metaverse: A survey. In: Companion Proceedings of the ACM Web Conference 2023. p. 1151–1160. WWW '23 Companion, Association for Computing Machinery, New York, NY, USA (2023), <https://doi.org/10.1145/3543873.3587584>
- [9] Cwierz, H., Díaz-Barrancas, F., Llinás, J.G., Pardo, P.J.: On the validity of virtual reality applications for professional use: A case study on color vision research and diagnosis. *IEEE Access* **9**, 138215–138224 (2021)
- [10] Deng, L.: The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine* **29**(6), 141–142 (2012)

- [11] Díaz-Barrancas, F., Flores-Martin, D., Berrocal, J.: Training a neural network on virtual reality devices: Challenges and limitations. In: 2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). pp. 787–788. IEEE (2024)
- [12] Ding, Y., Li, Y., Cheng, L.: Application of internet of things and virtual reality technology in college physical education. *Ieee Access* **8**, 96065–96074 (2020)
- [13] Dohan, M., Mu, M., Ajit, S., Hill, G.: Real-walk modelling: deep learning model for user mobility in virtual reality. *Multimedia Systems* **30**(1), 44 (2024)
- [14] Elor, A., Kurniawan, S.: Deep reinforcement learning in immersive virtual reality exergame for agent movement guidance. In: 2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH). pp. 1–7 (2020).
- [15] Flores-Martin, D., Berrocal, J., Garcia-Alonso, J., Murillo, J.M.: Towards a runtime devices adaptation in a multi-device environment based on people’s needs. In: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). pp. 304–309. IEEE (2019)
- [16] Flores-Martin, D., Galán-Jiménez, J., Berrocal, J., M. Murillo, J.: An analysis about federated learning in low-powerful devices. In: Proceedings of the 1st International Workshop on Middleware for the Computing Continuum. pp. 7–11 (2023)
- [17] Freina, L., Ott, M.: A literature review on immersive virtual reality in education: state of the art and perspectives. In: The international scientific conference elearning and software for education. vol. 1, pp. 10–1007 (2015)
- [18] Gheibi, O., Weyns, D., Quin, F.: Applying machine learning in self-adaptive systems: A systematic literature review. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* **15**(3), 1–37 (2021)
- [19] Khan, L.U., Saad, W., Han, Z., Hossain, E., Hong, C.S.: Federated learning for internet of things: Recent advances, taxonomy, and open challenges. *IEEE Communications Surveys & Tutorials* **23**(3), 1759–1799 (2021)
- [20] Li, L., Fan, Y., Tse, M., Lin, K.Y.: A review of applications in federated learning. *Computers & Industrial Engineering* **149**, 106854 (2020)
- [21] Li, T., Sahu, A.K., Talwalkar, A., Smith, V.: Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine* **37**(3), 50–60 (2020)

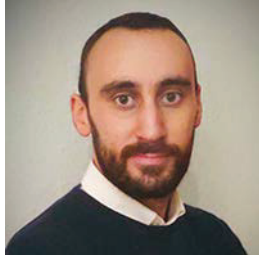
- [22] Mäkinen, H., Haavisto, E., Havola, S., Koivisto, J.M.: User experiences of virtual reality technologies for healthcare in learning: an integrative review. *Behaviour & Information Technology* **41**(1), 1–17 (2022)
- [23] Makransky, G., Borre-Gude, S., Mayer, R.E.: Motivational and cognitive benefits of training in immersive virtual reality based on multiple assessments. *Journal of Computer Assisted Learning* **35**(6), 691–707 (2019)
- [24] Malik, A.A., Masood, T., Bilberg, A.: Virtual reality in manufacturing: immersive and collaborative artificial-reality in design of human-robot workspace. *International Journal of Computer Integrated Manufacturing* **33**(1), 22–37 (2020)
- [25] Mevlevioğlu, D., Tabirca, S., Murphy, D.: Real-time classification of anxiety in virtual reality therapy using biosensors and a convolutional neural network. *Biosensors* **14**(3) (2024). , <https://www.mdpi.com/2079-6374/14/3/131>
- [26] Naraha, T., Akimoto, K., Yairi, I.E.: Survey of the vr environment for deep learning model development. In: Annual Conference of the Japanese Society for Artificial Intelligence. pp. 154–164. Springer (2021)
- [27] Qiao, D., Qian, L., Guo, S., Zhao, J., Zhou, P.: AMFL: Resource-efficient adaptive metaverse-based federated learning for the human-centric augmented reality applications. *IEEE Trans. Neural Netw. Learn. Syst.* **PP**, 1–15 (Jun 2024)
- [28] Rentero-Trejo, R., Flores-Martín, D., Galán-Jiménez, J., García-Alonso, J., Murillo, J.M., Berrocal, J.: Using federated learning to achieve proactive context-aware iot environments. *Journal of web engineering* **21**(1), 53–74 (2022)
- [29] Scavarelli, A., Arya, A., Teather, R.J.: Virtual reality and augmented reality in social learning spaces: a literature review. *Virtual Reality* **25**, 257–277 (2021)
- [30] Sharma, G., Chandra, S., Venkatraman, S., Mittal, A., Singh, V.: Artificial neural network in virtual reality: A survey. *International Journal of Virtual Reality* **15**(2), 44–52 (2016)
- [31] Xie, B., Liu, H., Alghofaili, R., Zhang, Y., Jiang, Y., Lobo, F.D., Li, C., Li, W., Huang, H., Akdere, M., et al.: A review on virtual reality skill training applications. *Frontiers in Virtual Reality* **2**, 645153 (2021)
- [32] Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* **10**(2), 1–19 (2019)

- [33] Zhang, T., Gao, L., He, C., Zhang, M., Krishnamachari, B., Avestimehr, A.S.: Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet of Things Magazine* **5**(1), 24–29 (2022)
- [34] Zhao, J., Yin, J., Shi, Y., Qiao, L., Ma, G.: User entertainment experience analysis of artificial intelligence entertainment robots based on convolutional neural networks in park plant landscape design. *Entertainment Computing* **52**, 100817 (2025)
- [35] Zhao, Y., Liu, S., et al.: A deep learning model with virtual reality technology for second language acquisition. *Mobile Information Systems* **2022** (2022)
- [36] Zhou, X., Liu, C., Zhao, J.: Resource allocation of federated learning for the metaverse with mobile augmented reality. *IEEE Transactions on Wireless Communications* (2023)

## Biographies



**Daniel Flores-Martin** is a researcher and systems and supercomputing administrator at the COMPUTAEX Foundation. His research interests include the Internet of Things, artificial intelligence, and high-performance computing.



**Francisco Díaz-Barrancas** is a post-doc researcher at the University of Extremadura. His main interests are virtual reality, artificial intelligence and color processing.



**Pedro J. Pardo** is an Associate Professor at the University of Extremadura. His research interests include color vision, neural networks and computer networks.



**Javier Berrocal** (IEEE Member) is an Associate Professor at the University of Extremadura. His main research interests are software architectures, mobile computing, and edge and fog computing.



**Juan Manuel Murillo** (IEEE Member) is a Full Professor at the University of Extremadura. His research interests include software architectures, mobile computing, and cloud computing.