International Journal of Internet, Broadcasting and Communication Vol.14 No.3 22-29 (2022)
http://dx.doi.org/10.7236/IJIBC.2022.14.3.22

IJIBC 22-3-4

A Novle Method for Efficient Mobile AR Service in Edge Mesh Network

Seyun Choi[1], Woosung Shim[2], Sukjun Hong[3], Hoijun Kim[4], Seunghyun Lee[5], and Soonchul Kwon[6*]

[1, 2, 3] MS Student, Department of Smart System, Kwangwoon University, South Korea
[4] PhD Student, Department of Plasma-Bio-Display, Kwangwoon University, South Korea
[5]Professor, Ingenium College Liberal Arts, Kwangwoon University, South Korea
[6]Associate professor, Graduate School of Smart Convergence, Kwangwoon University, South Korea*
*seyunchoi12@kw.ac.kr, sws5779@kw.ac.kr, 000jun2@kw.ac.kr, hoi97@kw.ac.kr, shlee@kw.ac.kr, *ksc0226@kw.ac.kr*

Abstract

Recently, with the development of mobile computing power, mobile-based VR and AR services are being developed. Due to network performance and computing power constraints, VR and AR services using large-capacity 3D content have limitations. A study on an efficient 3D content load method for a mobile device is required. The conventional method downloads all 3D content used for AR services at the same time. In this paper, we propose an active 3D content load according to the user's track. The proposed method is a partitioned 3D object load. Edge servers were installed for each area and connected through the MESH network. Partitioned load the required 3D object in the area referring to the user's location. The location is identified through the edge server information of the connected AP. The performance of the proposed method and the conventional method was compared. As a result of the comparison, the proposed method showed a stable Mobile AR Service. The results of this study, it is expected to contribute to the activation of edge server-based AR mobile services.

Keywords: *Augmented Reality, Cloud server, Edge server, Mesh, 3D content*

1. INTRODUCTION

In this paper, an experiment was performed to provide variant 3D AR content according to the connected MESH network. We partitioned the area into WI-FI Edge MESH networks. The experiment implemented variant Edge MESH networks on the 4th and 5th floors in the indoor environment. The internal network of each MESH network's configurated has equal WI-FI SSID, Password 2.4g, and WIFI channel. This method allows users to change the MESH network without reconnecting Wi-Fi. The environment was set up to automatically switch content corresponding MESH network. On the first connection, the client identifies the

Manuscript Received: May. 14, 2022 / Revised: May. 19, 2022 / Accepted: May. 22, 2022
Corresponding Author: kcs0226@kw.ac.kr
Tel: +82-2-940-8637, Fax: +82-50-4174-3258
Associate professor Graduate School of Smart Convergence, Kwangwoon University, South Korea

connected MESH network and provides suitable 3D AR content. Identification of the MESH network is used by reference PHP files from the edge server. It informs the connected MESH network. The client periodically checks the information of the file to recognize MESH network changes. When the user moves to another MESH network, the client detects MESH network changes. It loads the content suitable for the changed MESH network. In this case, content is downloaded from the MESH Edge server. We compared the loading time of 3D AR content between the edge server and the cloud server. The experiment compared the conventional method that downloads all 3D AR content on the first connection and the proposed method. Service was implemented in the VEEA Edge hub and EC2 of Amazon Web Services.

2. BACKGROUND THEORY

2.1 Wi-Fi Mesh & Edge Server

Wi-Fi Mesh was meant to improve network coverage. Network deployment is possible to provide Wi-Fi coverage without shadow[M계 1][M계 2] area. WMN (Wireless Mesh Network) is based on the use of mesh routers. The mesh router is configured with wireless interfaces, making it convenient to add and remove. The main aspect is to form a large Wi-Fi coverage with minimal interference. It is to cleverly assign different channels to each wireless interface [1-3]. Edge servers attract attention for their low latency and distributed processing. Due to the rapid development of the Internet of Things and 5G networks, a large amount of data is generated from smart cities [4]. As a result, the latency of cloud computing increases. An edge server is that propositioned to offload part of the workload from the mobile edge computing device to a peripheral edge server with sufficient computational resources to reduce latency [5]. Edge computing is an approach to bringing computing resources closer to end users [6]. In this paper, edge servers and cloud servers were tested in various environments. When using an edge server, there are fewer network problems and AR content is delivered quickly [7]. Figure 1 shows the diagram of the Wi-Fi Mesh and Edge Server.

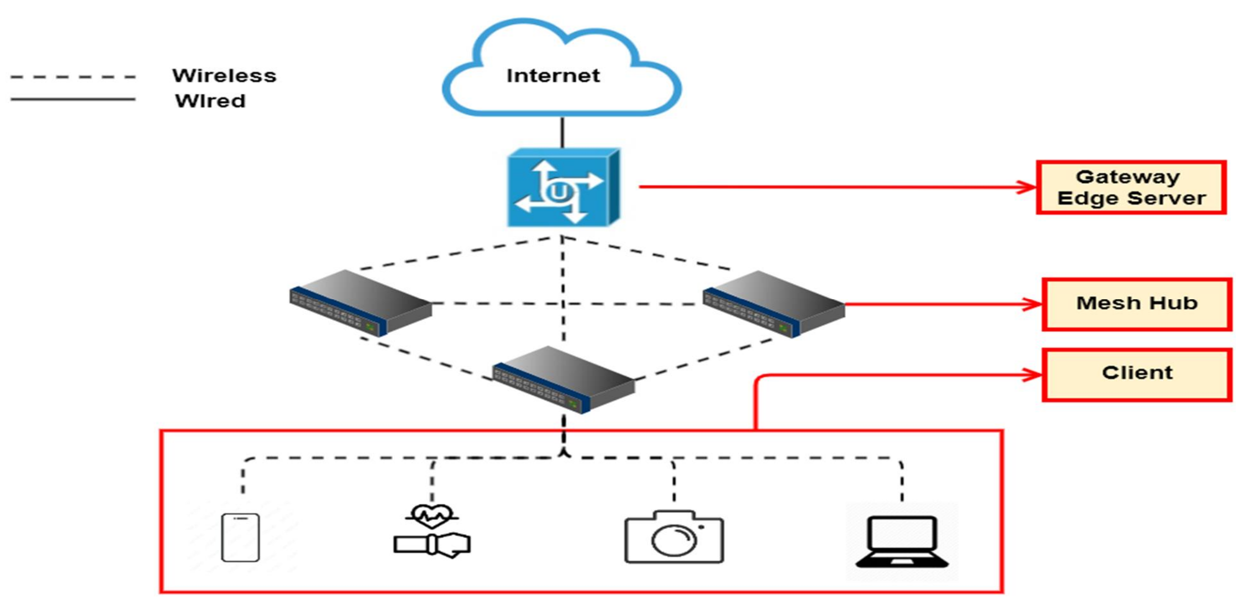


Figure 1. Wi-Fi Mesh & Edge Server Structure

2.2 3D Formats

3D file formats typically include OBJ, STL, and FBX. Each format has limitations depending on its composition and uses [8]. Supported formats differ by a software program. The glb is organized in a binary format that helps the uniform integration of large 3D file formats on the web. A form that can transmit 3D geometry files more efficiently is glb. One of the basic elements of glb is a JSON file that specifies the structure or composition of the scene containing the 3D model. Figure 2 shows illustrate the concept of glb-related top-level elements. It is the Camera that sets the viewpoint of the Scene and Node, which are the most basic structures. Mesh represents the geometric information of 3D objects to be imported into the web. Buffer View and Accessor represent reference and layout-related information.

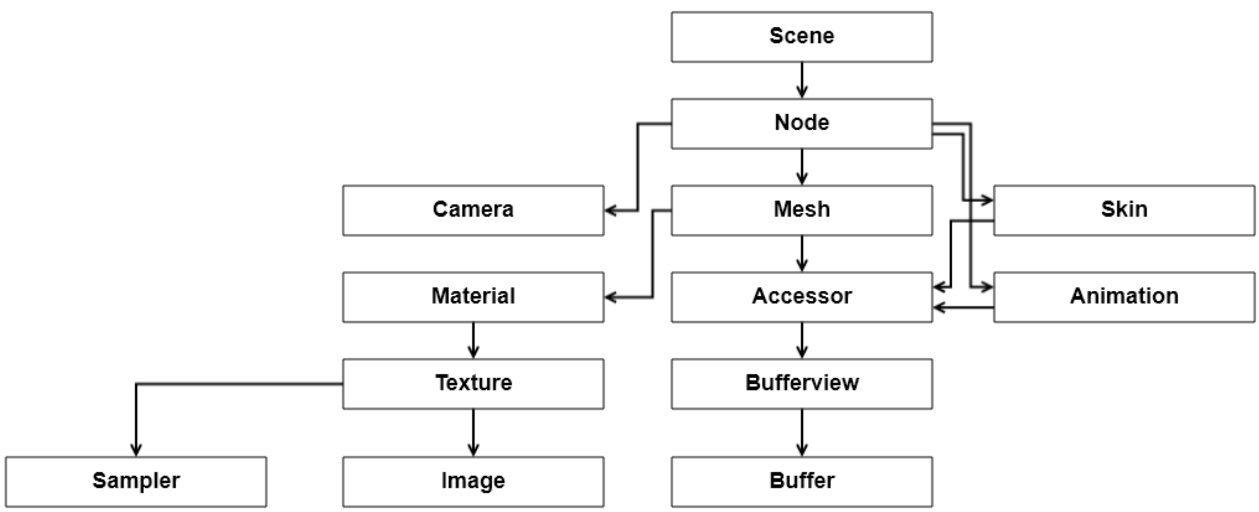


Figure 2. glb Format Structure

According to research related to the glb format, as shown in Table 1, glb supports important features such as colors, animations, CSG (Constructive Solid Geometry), detailed meshing, textures, cameras, lighting, and relative positions [9].

Table 1. Various 3D Formats Comparison

File Type	Geometry		Appearance					Scene		Animation
	Brief Mesh	Detailed Mesh	CSG	Color	Material	Texture	Camera	Light	Relative Position	
STL	O	X	X	X	X	X	X	X	X	X
OBJ	O	X	O	O	O	X	X	X	X	X
FBX	O	X	O	O	O	O	O	O	O	O
glTF (glb)	O	O	O	O	O	O	O	O	O	O

3. EXPERIMENTS

Implementing different MESH networks on the 4th and 5th floors, a system was established that provides different content on the 4th floor and 5th floor. 3D content called 4floor.glb provided on the 4th floor and

5floor.glb on the 5th floor. Each mesh network has a floor.php file that can identify MESH network information, and the file has floor information suitable for each MESH network. The client detects that the MESH network has switched by periodically checking the floor.php file. When the MESH network is changed, the content of the changed MESH network can be prepared. Figure 3 shows the glb file model used in the experiment.

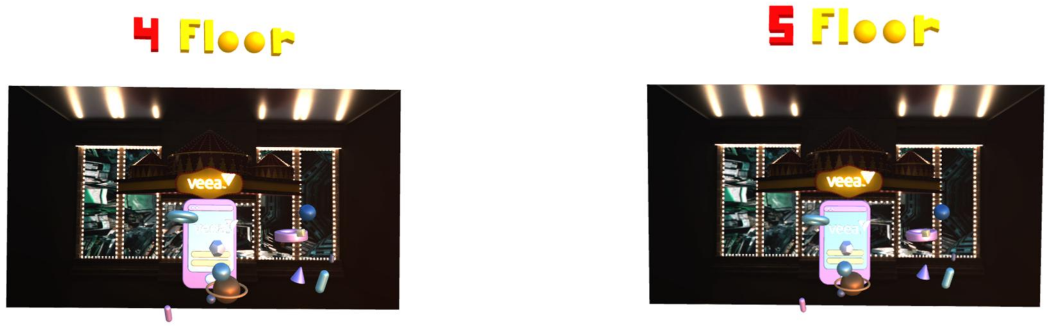


Figure 3. AR Content Used in The Experiment

We used VHE10 (VEEA hub) for MESH network construction. The configuration is as follows. VHE10 for the web server and AP are placed on the 4th and 5th floors, respectively. Edge server was mounted on VHE10 installed for MESH network establishment. For the cloud server to compare with the edge server, it was built with Amazon Elastic Compute Cloud (EC2) of amazon web services. Figure 4 shows the network configuration.

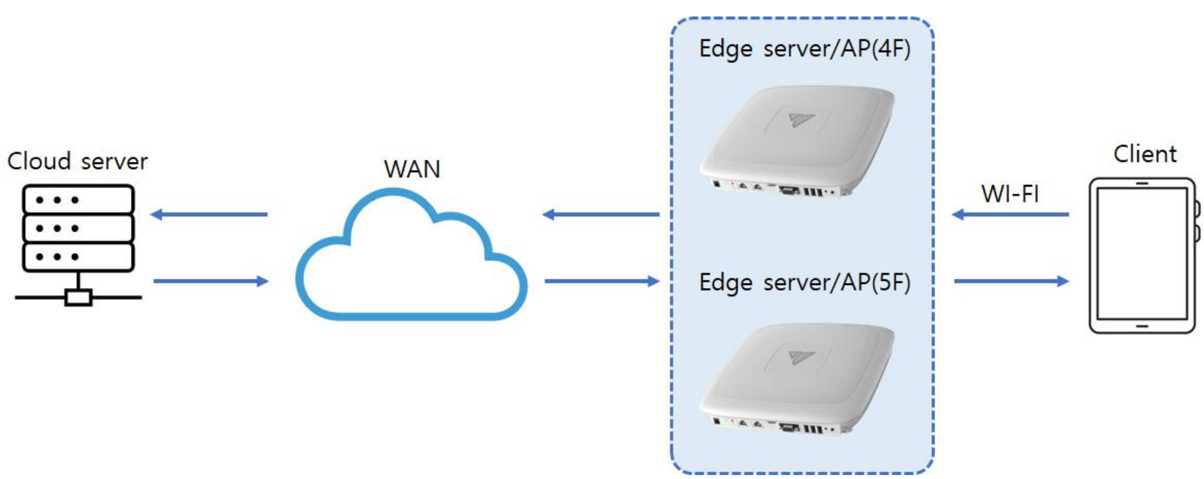
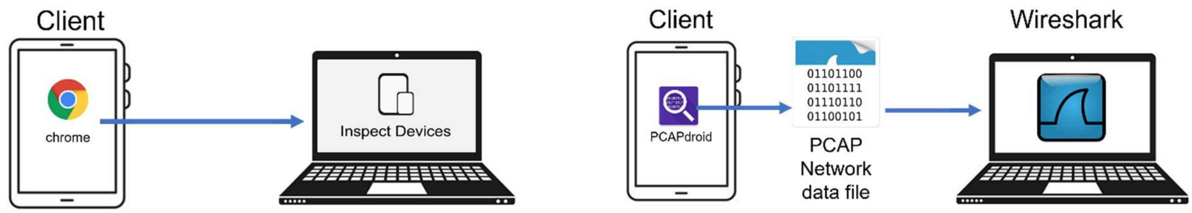


Figure 4. Network Configuration for Experimentation

The experiment measured the time it takes to complete the download of 3D content 4floor.glb and 5floor.glb. To record the measurement results, a mobile device for data collection and a laptop were connected via USB, and the network of Chrome device inspection was monitored in real-time. Client devices used the PCAPdroid app to perform packet capture and output to PCAP file format. Real-time web traffic was measured by analyzing PCAP format files with I/O Graphs using Wireshark. Figure 5 shows the data collection and analysis method. Figure 5 (a) is Chrome Inspect Analysis used to collect 3D content download time. Figure 5 (b) is Wireshark Analysis that collects Real-time I/O Interface communication speed.



(a) (b)

Figure 5. Experimental Results Data Collection and Analysis Workflow
(a)Chrome Inspect Analysis (b) Wireshark Analysis

3.1 EXPERIMENTS ENVIRONMENT

Table 2 shows the server environment used in the experiment. Table 3 shows the environment of the client used for the experiment. Table 4 shows the computer environment used for the analysis.

Table 2. Server Environment

	Edge Server (VEEA VHE10)	Cloud Server (Amazon EC2)
Type	-	t2.micro
CPU	ARMv8 Quad Core processer, 1.5GHz	vCPUs, 2.5 GHz
RAM	8GB DRAM	1G
Wi-Fi	Tri-band Wi-Fi5	-
NETWORK	1Gbps	-
OS	Ubuntu 20.04 (AMD 64bit)	Ubuntu 20.04 (AMD 64bit)
Apache2	2.4.41	2.4.41
php	7.4.3	7.4.3

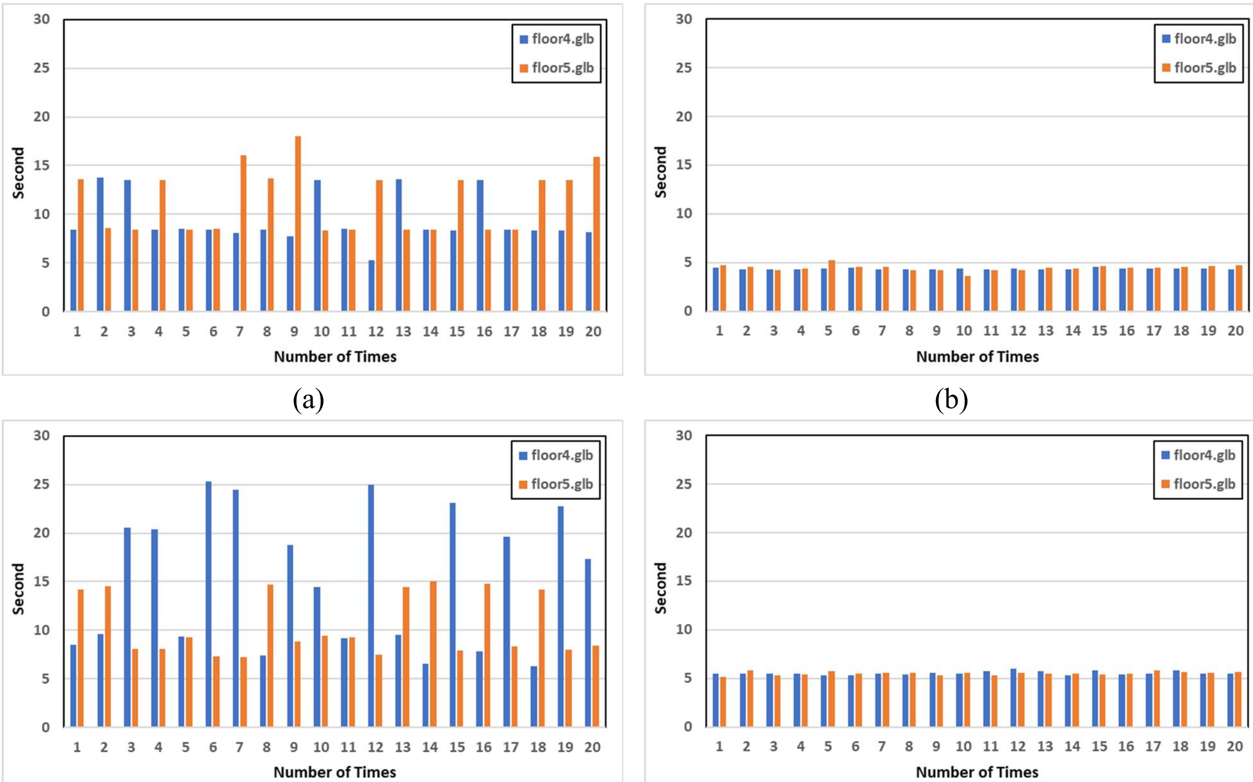
Table 3. Client Environment

Client (Samsung tab S6 SM-T860)	
CPU	Qualcomm Snapdragon 855 SM8150 Platform.
RAM	8 GB LPDDR4X SDRAM
OS	Android 11
Wi-Fi	Wi-Fi 1/2/3/4/5
Chrome	97.0.4692.98
PCAPdroid	1.4.5

Table 4. Analysis Computer Environment

Computer for analysis (Samsung Galaxy Book ion 2 NT950XDZ-A58AW)	
CPU	11th gen intel(R) core(TM) i5-1135g7
RAM	16 GB
OS	11 Pro insider preview build : 22538.rs_prerelease.220114-1500
Chrome	97.0.4692.99
Wireshark	3.6.1

3.2 EXPERIMENTS RESULT.



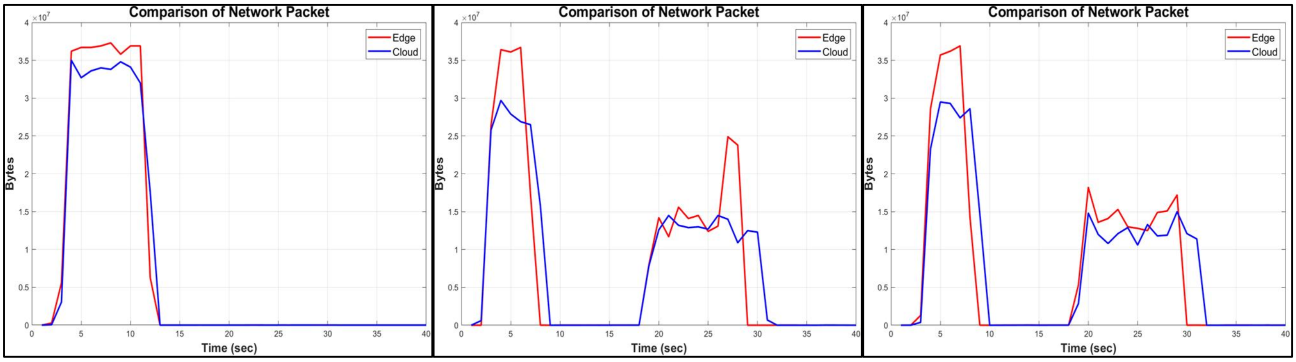
(a) (b) (c) (d)

Figure 6. Comparison of 3D Content Download Time between The Proposed Method and The Conventional Method

(a) Conventional Method based Edge (b) Proposed Method based Edge

(c) Conventional Method based Cloud (d) Proposed Method based Cloud

Figure 6 shows the results of 3D content download time measured by Chrome Inspect. In figure 6 (a) and figure 6 (c), the experiment was conducted by downloading 3D content at once when accessing the web browser for the first time. Figure 6 (b) and figure 6 (d) is an experiment results that downloads only required 3D content for each MESH network. When connecting for the first time on the 4th floor, only 4floor.glb is downloaded, and when the MESH network is moved to the 5th floor, 5floor.glb is downloaded. Figure 6 (a) and (b) are the content of the experiment on the edge server, and figure 6 (c) and (d) are the content of the cloud server. The average standard deviation of edge servers in figure 6 (a) and (b) is 2.90 and 0.18, which is more stable than the standard deviations of 5.0, and 0.19 of cloud servers in figure 6 (c) and (d). The average standard deviations of the proposed method in figure 6 (b) and figure 6 (d) are 0.18 and 0.19; the average standard deviations of the conventional method in figure 6 (a) and figure 6 (c) are 2.90 and 5.0. The proposed method showed a more stable result than the conventional method.



(a) (b) (c)

Figure 7. Real-time I/O Interface Communication Speed
(a) Conventional Download Method
(b) Proposed Download Method; Client Movement 4th floor to 5th floor Load
(c) Proposed Download Method; Client Movement 5th floor to 4th floor Load

Figure 7 shows the amount of communication speed of the real-time I/O interface analyzed by Wireshark. Figure 7 (a) is the result of measuring how to download 3D content at once when connecting to a web browser for the first time. Figure 7 (b) is the result of the client starting on the 4th floor and downloading the 4floor.glb initially, moved to 5th floor and MESH network changes, download 5floor.glb later. In case figure 7 (c) is the result of the client starting on the 5th floor and downloading 5floor.glb initially, moved to 4th floor and MESH network changes, download 4floor.glb later. The initial load time of the Proposed Method figure 7 (b) and figure 7 (c) is about 50% faster than that of the conventional method figure 7 (a). Also, the result of downloading from the edge server shows a slightly faster download speed than the result of downloading from the cloud server.

4. CONCLUSION

Edge server shows better performance in communication speed and stability than a cloud server. Edge server communicates through internal network communication. It is not affected by external network conditions and has fewer physical hops. As a result, network variables are reduced, and the results are more stable and faster than the cloud server. The proposed method that downloads the suitable content for each MESH network has shown better network stability than the conventional method. In the conventional method, the client receives a large amount of various data such as HTML, js, and whole 3D AR content from the server on the first connection. It occurs with large initial network traffic. The proposed method only initially downloads the HTML, js, and necessary 3D AR content. It avoids large network traffic in the initial download. When the connected MESH network changes, the file is additionally downloaded. It causes distributed download traffic that ensures stable downloads. In addition, unnecessary data that is not used is not downloaded by the client which can reduce unnecessary traffic. As a result, the delay time when accessing the initial website is reduced and download stability is increased. Through this research, we expected that it will be able to provide a faster and more stable indoor Web AR service. In the future, more efficient mesh segmentation methods and various preload methods should be studied.

Acknowledgement

This research is supported by Ministry of Culture, Sports and Tourism and Korea Creative Content Agency (Project Number: R2021040083)

References

[1] R. Riggio, T. Rasheed, S. Testi, F. Granelli, and I. Chlamtac, “Interference and traffic aware channel assignment in WiFi-based wireless mesh networks,” Ad Hoc Networks, vol. 9, no. 5. Elsevier BV, pp. 864–875, Jul 2011. DOI: https://doi.org/10.1016/j.adhoc.2010.09.012

[2] V. Sevani, B. Raman, and P. Joshi, “Implementation-Based Evaluation of a Full-Fledged Multihop TDMA-MAC for WiFi Mesh Networks,” IEEE Transactions on Mobile Computing, vol. 13, no. 2. Institute of Electrical and Electronics Engineers (IEEE), pp. 392–406, Feb 2014. DOI: https://doi.org/10.1109/TMC.2012.251.

[3] N. B. Salem and J.-P. Hubaux, “Securing wireless mesh networks,” IEEE Wireless Communications, vol. 13, no. 2. Institute of Electrical and Electronics Engineers (IEEE), pp. 50–55, Apr 2006. DOI: https://doi.org/10.1109/MWC.2006.1632480

[4] M. Marjani, F. Nasaruddin, A. Gani, A. Karim, I. A. T. Hashem, A. Siddiqa, and I. Yaqoob “Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges,” IEEE Access, vol. 5. Institute of Electrical and Electronics Engineers (IEEE), pp. 5247–5261, 2017. DOI: https://doi.org/10.1109/ACCESS.2017.2689040

[5] S. Wang, Y. Zhao, J. Xu, J. Yuan, and C.-H. Hsu, “Edge server placement in mobile edge computing,” Journal of Parallel and Distributed Computing, vol. 127. Elsevier BV, pp. 160–168, May 2019. DOI: https://doi.org/10.1016/j.jpdc.2018.06.008

[6] B. Charyyev, E. Arslan, and M. H. Gunes, “Latency Comparison of Cloud Datacenters and Edge Servers,” GLOBECOM 2020 - 2020 IEEE Global Communications Conference. IEEE, Dec 2020. DOI: https://doi.org/10.1109/GLOBECOM42002.2020.9322406

[7] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, “Delay-optimal computation task scheduling for mobile-edge computing systems,” 2016 IEEE International Symposium on Information Theory (ISIT). IEEE, Jul 2016. DOI: https://doi.org/10.1109/ISIT.2016.7541539

[8] G. Lee, J. Nam, H. Han, and S. Kwon, “A Study on 3D File Format for Web-based Scientific Visualization,” International Journal of Advanced Culture Technology, vol. 7, no. 1, pp. 243–247, Mar 2019. DOI: https://doi.org/10.17703/IJACT.2019.7.1.243

[9] G. Lee, P. Choi, J. Nam, H. Han, S. Lee, and S. Kwon, “A Study on the Performance Comparison of 3D File Formats on the Web,” International journal of advanced smart convergence, vol. 8, no. 1, pp. 65–74, Mar 2019. DOI: https://doi.org/10.7236/IJASC.2019.8.1.65