Short Survey on Computing in Real Time Mixed Reality

Hanson Tran

Indiana University – Purdue University Indianapolis

CSCI 44300

Database Systems – Kelly Van Busum

04/29/2022

Abstract

Mixed reality (MR) is a technology that faces numerous issues during the design and implementation phases, particularly those related to time-sensitive, real-time applications. The primary goal of this survey is to present a conceptual model for MR applications to connect and interact with 5G devices in a 3D space. The conceptual model is based on utilizing individual voxels to offload computational costs.

Keywords: *Mixed Reality, Video Conferencing, Voxel, Database Visualization*

Contents

1. INTRODUCTION

1.1. Narrative

The desire to work from home has grown in recent years, particularly considering the Covid restriction. With the advancement of 5G and other edge computing technologies and devices, businesses should provide a collaborative experience in mixed reality for their employees and customers alike for planning, designing, debugging, or instructing using edge computing. For example, user A could take a few photos of an existing object or physical space, have that data modeled to a 1:1 scale in CAD, and then have other users work in real time on further developing that product or working on that object in a replicated virtual reality setting. This information is then relayed back to user A via augmented or virtual reality. Think of it like a more immersive Zoom that could be useful for things like planning, design, debugging, and instructing.

1.2. Preface

With over 4.5 billion active internet users, including a growing number of smartphone users [5,] it is critical that we can bridge communication gaps to expand business and trade. In its current state, mixed reality is not used in the remote conferencing space. Current technologies use cloud storage to host video streams from individual users. A service like Zoom, for example, would process all video feeds sent from users at a central server, and the central server would then transcode video data to parameters that meet user standards. This could include a variety of standards like high resolution, low compression artifacts, low latency, low bandwidth costs, and low CPU costs. Mixed Reality is typically classified as one of the following: virtual reality predominantly in gaming and simulation and augmented reality predominantly for education or communications. Not many systems offer a combined experience for remote collaborative design in a mixed reality space. Microsoft is one of the few companies to offer such a service something like this with their Azure mixed reality and HoloLens. However, their service would limit the ability for us to leverage machine-to-machine services in the growing 5G space.

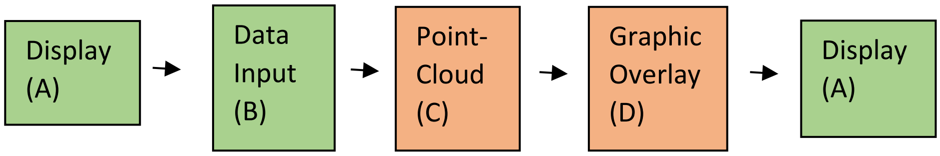
Current literature suggests that databases are handled in predominately traditional ways. Suggesting employing two main databases. One database for 3D part information and one separate database for all other information. As 5G machine-to-machine devices become more readily available, it is important to consider how to revisualize the SQL database.

Through my review of current research literature, I discovered that voxels can be used to offset database costs. A voxel is a 3D cube that represents a pixel [4]. I believe that with the help of expanding 5G networks, some of the database costs can be offloaded to each user if a flexible framework is developed. The framework I'll be discussing today is a mash-up of techniques discovered during my survey.

1.3. Contents

This paper will be divided into five sections. The first four sections will simulate the system's process flow. Figure 1.1 depicts this process flow. The final body section will go over the potential costs and scalability of my framework as well as proof methods. The procedure is as follows: The user first launches a Unity-created application (A). Data tables or CAD information are used to manage data input for specific objects (B). The Point-Cloud technique is used to collect generic background information for the current scene in physical space (C). Background information and specific object information are further processed in the graphic overlay, along with video and audio feed (D), before being displayed back to the user (A). In the process flow chart, processes in green are of low computational costs and those in orange are of higher computational costs.

Figure 1.1



2. Display

Throughout my survey, Unity and Steam VR were frequently mentioned as tools used by researchers to create mixed reality applications. Both tools provide user flexibility in terms of device hardware and VR hardware. The Unity engine is adaptable in the sense that it allows designers to create interfaces in 3D space using built-in tools or C#. Unity also promotes its cloud capabilities.

2.1 User Interface

When opening the framework application from a 5G capable device, the user is met with a generic screen with a multitude of options. These options range from inputting data, modifying data, rendering scene, to live collaboration with audio and video feed. The user interface is the most flexible portion of this framework. Throughout the framework, information will go from the user to server and back. The design here is flexible as the desired user interface can be created within Unity.

3. Raw data input

The framework starts by loading in data through a 5G smartphone or another machine-to-machine capable device. The issue lies in the fact that different devices will have different hardware installed. The framework this survey introduces attempts to mitigate some of those differences through an array of preprocessing techniques.

3.1 Pre-Processing

Most of the preprocessing will be managed by taking in the raw data and meta data of data tables and converting them into a voxel space where each voxel cube represents a single table. This process is part of the VRiDaM solution framework introduced by researchers from Aalen University in Germany. The VRiDAM framework supports SQL and has protocols in place to manage MSSQL, Mongo, PostgreSQL, Cassandra, and Neo4 database management systems [1]. The ability to preprocess the data and it immediately be in a 3D space to use is a tremendous improvement over a traditional cloud database system. Current traditional systems attempt to take geospatial information and overlay them on top of the physical space [6]. The issue with this is that most of the costs are still offloaded by the datacenters, not through growing 5G devices. Computational costs for these models also involve the use of a CNN to align the virtual realm and the physical realm. This reoccurring computation can be extremely costly. The voxel as defined in this framework can hold all database table information. Users can access the face corresponding to the datatype they wish to manipulate, and all related information is passed on to the cloud. This means that the voxel proposed can handle geographic information, asset information, and all-encompassing cases introduced by the users.

3.2 Processed Voxel

After the data is processed, it is possible to begin utilizing it in the 3D space. This is because each individual voxel or cube is representative of a database table. The cube is powerful in that each of its 6 faces can be utilized for different purposes a normal SQL database would see. It is possible to freely group the cube or combinations of cubes based on this. Relations with different tables can be done through a visual network of pipes (lines in the 3D space), attributes are displayed by behind the original cube as an attached cube to one of its faces. The voxel is now ready for the user to manipulate [1]. The user can manipulate each voxel according to the UI designed within the Unity engine. Tools like inputting data, modifying data, rendering scene, to live collaboration are all possible tools in the modified framework suggested in this survey.

4. Point Cloud

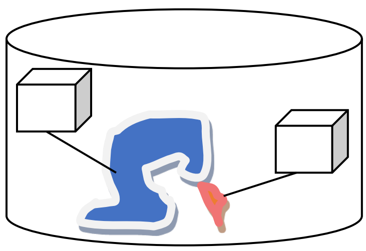
A point cloud would be a set of data points in each space. This finds its uses within our framework in two main areas: the point cloud allows for real-time troubleshooting as well as allows for helpful modeling tool for its users. This helps offset the need for everything to be modeled in CAD and converted into voxels as the general background can be scanned in and specific object models handled through CAD. One computationally heavy portion of this framework is of this CAD conversion process. There will be a CNN in place that can help convert CAD files to 3D voxel files.

4.1. Point Cloud to Voxel

The main issue with frameworks found throughout my survey is that most of its users will be utilizing specific equipment. With camera limitations and different camera hardware for different systems, it was important to identity ways to convert a 2D photo into a 3D space [2]. The proposed framework in this survey proposes two separate ways to handle this. The first technique is to have a strong enough neural network model to segment the incoming image in a series of bounding boxes and predict object segmentation and classification on it. This is troubling because this proposal is both computationally intensive and the models that work well in classifying and recognizing certain objects may struggle to do so for unrelated objects in its training data set. The secondary method is preferred but comes with higher costs for the user. Instead of predicting objects in a certain distance, the framework instead segments everything within a given space and splits each segmented object into its own clusters of voxels, with its core voxel representing a generic data table to represent an unidentified object. This is a modified approach of a Benediktine visualization technique [7]. The original technique maps an object's attributes to its intrinsic and extrinsic values.

Here, because the scene has yet to be built, users can scan the surroundings as a panoramic photo that is stored as a cylindrical object that can later be separated into individual object voxel clusters. This gives users the ability to remove clusters from the background scene cylinder to be modeled through CAD techniques. Figure 1.2 below shows an idea of what is happening. The panoramic background is represented by the cylinder. The two cubes are representative of two separate users, one who is modeling the object in blue and one the object in red. Once complete, the panoramic is stored as one object and the modeled objects are stored as another, with distance parameters to panoramic object stored.

Figure 1.2



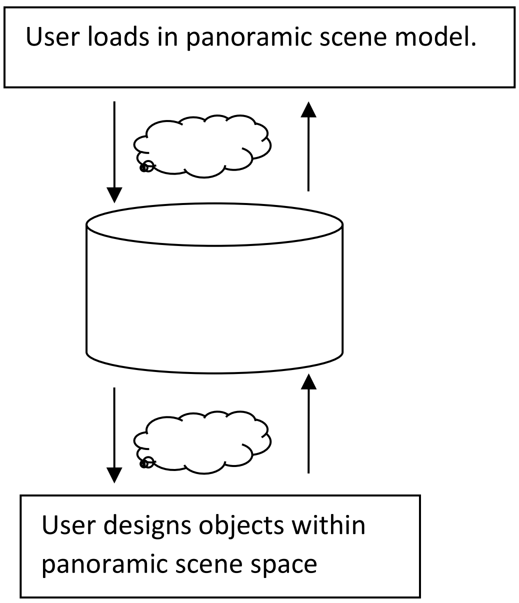
5. Graphic Overlay

The graphic overlay is one of the most computationally heavy portions of this survey's framework. This is where model and scene information is relayed back and forth between user and cloud servers. Traditional frameworks have the 3D CAD design data saved onto the cloud through a series of restful APIs. These API callouts store each object with a unique ID and goes through a series of recursive functions to determine X, Y, and Z axis location of objects and backgrounds. This information is then voxelized for use [3].

5.1 Offloading Costs

Through my survey, there is a way to offset costs with the use of the of the voxel and point cloud system mentioned earlier. Because the physical space is stored as a single cylindrical voxel, that single object can be sent and retrieved from the cloud. Within this cylindrical voxel space, users can build voxel objects as they please. These object locations are referenced to the cylindrical model space easily as every cube within the object represents a single unit of measure in the real space. These means that rather than a series of APIs to build and update the scenery and models within, users instead are only tasked with modeling relevant objects within a scene. This also means that these costs can be offloaded to individual users through their 5G devices. This is because each user is not expected to render the entire scene on their device, nor is the cloud center expected to. Instead, each 5G device is tasked with rendering the single model the user is interacting with and referencing the cloud database for location references to the scene cylinder voxel. Figure 1.3 below shows how users interact with the cloud. Each interaction calls the cloud network to references distance parameters from created objects to the panoramic scene model. This is better than offloading costs to the cloud network in the form of calls to the restful API and having the CNN compute distance parameters. Here, the distance coordinate relative to the cylindrical voxel space is referenced continuously. While initial loading costs of the databases (voxels) are higher, this can help improve accuracy as the cylindrical space is mapped at a 1:1 scale to the physical space.

Figure 1.3



5.2 Video and Audio Conferencing

Because the framework calls for the use of voxels as potential databases. As stated previously within this survey, it has been found that a cube can represent a data table with each side representing a different attribute within the data table. If each user is represented as a cube with one face representing a static icon or a video stream, one cube face linked to an audio stream, and one cube face linked to keys and unique identifiers needed to access the cloud database, then it is then possible to video conference such as a normal Zoom call would, however you are now set in the 3D space. Computational costs would be cheapest if users only utilize an audio stream and utilize a static icon instead of a live video stream. This is because the cloud server would not have to receive the video feed, transcode the video feed, then relay the video feed back to other users. Listed in Table 1.1 below are the traditional bandwidth costs of a Zoom stream as stated by Zoom for a group call [8].

Table 1.1

Quality	Download	Upload	Total
720p	675 MB/hr.	675 MB/hr.	1.35 GB/hr.
1080p	1.2 GB/hr.	1.2 GB/hr.	2.4 GB/hr.

If we compare video streaming costs to that of an audio stream's 72 MB/hr., it makes much more sense to have a static icon than a video stream. However, for conferencing purposes like a meeting or a classroom, it is possible for users to switch on their video stream as needed.

6. Scalability, Security, Testing, and Conclusions.

The proposed framework in this survey attempts to offset costs from the traditional cloud network with the use of expanding 5G networks and devices. The proposed conclusion from my survey of current research literature is that each 5G device be responsible for hosting a user voxel that is capable of rendering and manipulating small object components within a set panoramic cylindrical voxel space.

6.1 Scalability

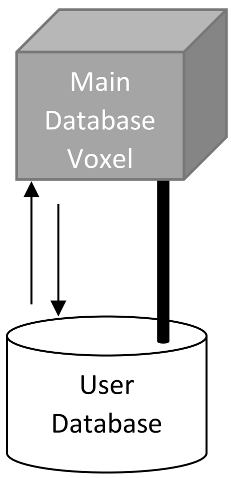
As the framework is dependent on utilizing 5G devices to offset computational costs from the cloud server, user costs are predominantly in acquiring their 5G device such as a smartphone and a VR headset. Additional service costs would be determined by the individual cloud services. Cost and performance are expected to scale along with the growth of 5G devices leveraging the growing micro centers with multi-access edge enabled. Uptime Institute references 5G as part of the reason for half of all workloads being ran outside of enterprise data as of 2021 [9]. 5G machine-to-machine transmit speeds can reach 20 Gbps and is up to IPv6 protocol [10]. This ensures that user computation on their own local 5G devices is possible.

6.2 Security

As stated previously within this survey, it is possible to designate a unique identifier for each user cube voxel. This allows the user voxel to have a multitude of representative identifiers that ensure security. Zoom operates using transport encryption. That is the same way our web browser connections work. Zoom operates on a combination of TCP and UDP protocols. The TCP connections use TLS similarly to how our web browsers interact with the rest of the internet. Following such a protocol would ensure a secure connection between users and the cloud network for video and audio feeds.

This framework also ensures the security of the model voxels of the panoramic scene and specific object voxels by ensuring that only users with specific keys have access to modeling functions. Users without access to modeling functions are still capable of listening to the stream like a normal conference call. The only difference the proposed framework in this survey has from a normal SQL database system is that rather than a 2D display of each data table, the data table is that each voxel cube can represent an entire data table. This means that all the security ensured within a normal SQL database is offered within this potential framework as well. Figure 1.4 below demonstrates how the user established databases interact with the cloud servers.

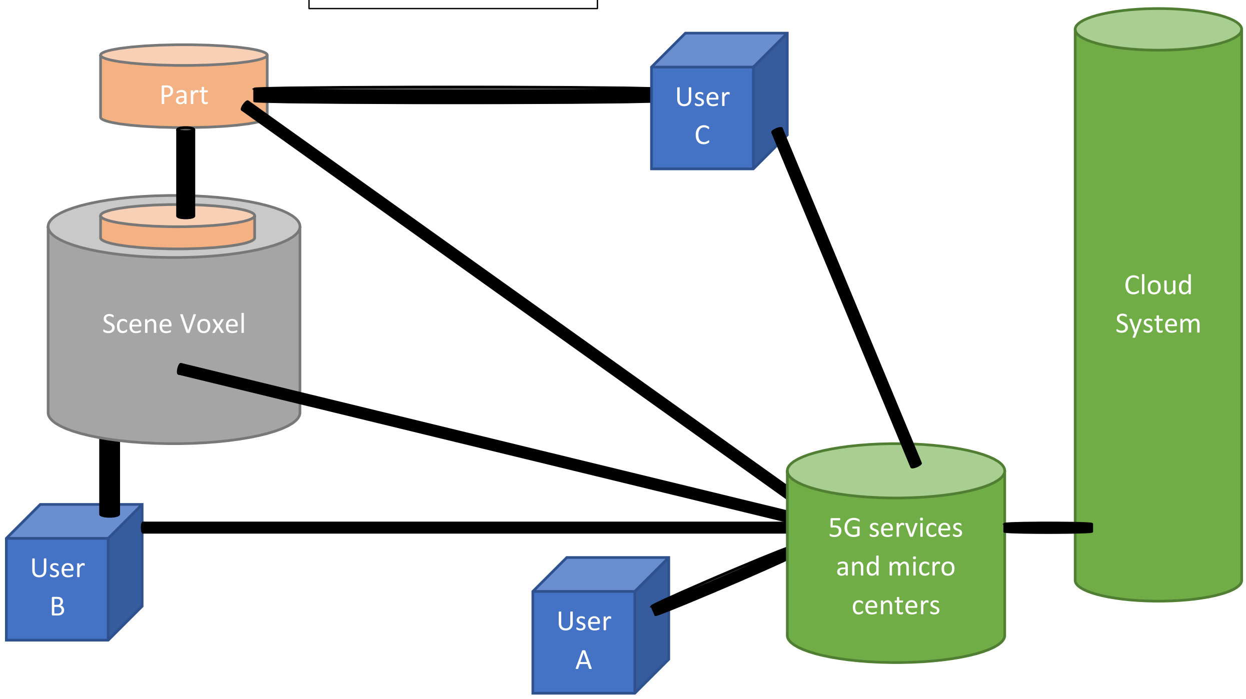
Figure 1.4



User database voxel has access to main database voxel through the pipeline if the User's unique information has access according to the main database. This can follow current SQL rules as well as TCP and UDP connections to ensure TLS standards are met.

Figure 1.5 below shows an illustration of how certain parts of the database can be made available to the users to.

Figure 1.5



As displayed by figure 1.5 above, user voxel A is a viewer, unable to interact with any design elements. User B has access to the scene voxel. All changes made by user B will be displayed to all, relayed by 5G to the cloud. User C has access only to the part voxel. Changes made by User C is displayed to all, relayed by 5G to the cloud.

6.3 Testing

It is difficult to quantify how much cost this can offload from the cloud server as the current literature does not state computational speeds, but rather quantify success in the form of fps rates and student surveys. The literature from the researchers at Aalen University suggests that while cube voxels are ideal in fps rates, you can utilize other shapes. They also found that a voxel-based system was an accurate system for students to use, albeit more difficult to initially learn on as current 3D learning platforms are limited. The Aalen University researchers conducted a survey to measure accuracy with their voxel-based system. Their system was tested on computer science students with no prior knowledge of SQL databases. They were asked varying questions on table relations between voxels, which table had certain column attributes, and what foreign and primary keys were available to each voxel cube. They showed a 92% accuracy on their web browser application and a 95% on their stand-alone application built with Unity. The web browser test group were asked 130 questions over 5 tasks, getting 6 wrong, and missing 4 on average. The group that tested their standalone application had 140 questions over 5 tasks, getting 1 wrong, and missing 6 on average.

Touting accuracy and ease of use, you can begin to scale up speeds and remain accurate. The main goal of this framework is to have as much of the computational cost be offset to each user, in hopes of allowing the growing 5G machine-to-machine connections and growing micro servers further offset that cost. You can quantify the success of this framework from packet data sent to and from the users to the cloud network.

6.4 Conclusions

While this framework is based on an amalgamation of different techniques found within my research of current literature, I believe there is a strong possibility of offloading computational costs from cloud networks using voxels in terms of mixed reality collaboration. By utilizing the voxel as a potential database as well as a source for object modeling, the computational costs are offset to the individual 5G devices and micro servers nearby. The framework proposed here allows users to communicate, collaborate, and utilize existing cloud frameworks as well as the growing 5G networks.

7. References

[1] R. Oberhauser, "Database Model Visualization in Virtual Reality: Exploring WebVR and Native VR Concepts." Accessed: Feb. 15, 2022. [Online]. Available: https://opus-htw-aalen.bszbw.de/frontdoor/deliver/index/docId/980/file/IJASW19-VR-DB_OberhauserFinal.pdf..

[2] D. Katare and M. El-Sharkawy, "Real-Time 3-D Segmentation on An Autonomous Embedded System: using Point Cloud and Camera," 2019 IEEE National Aerospace and Electronics Conference (NAECON), Jul. 2019, doi: 10.1109/naecon46414.2019.9057988.

[3] D. Mourtzis, J. Angelopoulos, and N. Panopoulos, "Collaborative manufacturing design: a mixed reality and cloud-based framework for part design," Procedia CIRP, vol. 100, pp. 97–102, 2021, doi: 10.1016/j.procir.2021.05.016

[4] Regenbrecht, H., Park, J., Ott, C., Mills, S., Cook, M. and Langlotz, T., 2022. *Preaching Voxels: An Alternative Approach to Mixed Reality*.

[5] Jaakonmäki, R., Müller, O., & Vom Brocke, J. (2017). The impact of content, context, and creator on user engagement in social media marketing. *Proceedings of the 50th Hawaii International Conference on System Sciences (2017)*. doi:10.24251/hicss.2017.136

[6] Talmaki, S., Dong, S. and Kamat, V., 2022. *Geospatial Databases and Augmented Reality Visualization for Improving Safety in Urban Excavation Operations*.

[7] R. Oberhauser, "Database Model Visualization in Virtual Reality: A WebVR and Benediktine Space Approach," In: Proceedings of the Thirteenth International Conference on Software Engineering Advances (ICSEA 2018). IARIA, pp. 108-113, 2018

[8] 2022. [online] Available at: <https://support.zoom.us/hc/en-us/articles/201362023-Zoom-system-requirements-Windows-macOS-Linux#h_d278c327-e03d-4896-b19a-96a8f3c0c69c> [Accessed 29 April 2022].

[9] Rhonda Ascierto, U., 2022. Edge Computing - The Next Frontier - Uptime Institute Blog. [online] Uptime Institute Blog. Available at: <https://journal.uptimeinstitute.com/edge-computing-the-next-frontier/> [Accessed 28 April 2022].

[10] Dangi, R., Lalwani, P., Choudhary, G., You, I. and Pau, G., 2022. *Study and Investigation on 5G Technology: A Systematic Review*.