ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XII-4/W1-2026
21st International 3D GeoInfo Conference 2026, 28–29 September 2026, Sofia, Bulgaria

Exploring Point Cloud Interaction and AI Features for Sewer System 3D-Scans

Markus Brøndberg Nielsen[1], Lasse Hedegaard Hansen[2], Ivan Nikolov[3], Lars Bodum [2]

[1] Department of Architecture, Design and Media Technology, Aalborg University, Aalborg, Denmark - markun21@student.aau.dk
[2] Department of Sustainability and Planning, Aalborg University, Aalborg, Denmark - (lhhan, lbo)@plan.aau.dk
[3] Department of Architecture, Design and Media Technology, Aalborg University, Copenhagen, Denmark - iani@create.aau.dk

Keywords: point cloud viewer, sewer systems, Large Language Model (LLM), interactive application, visualization

Abstract

Visual analysis of point clouds from Laser Scanning of complex structures, especially those scanned from the interior, can be a complex, multi-step process due to intricate geometries, overlapping surfaces, and the relatively small size of internal components. Large underground sewerage systems are often surveyed using Laser Scanning to document a complete 3D scan of the interior. In such cases, visual analysis of the 3D point cloud can be very useful for detecting problems, planning repairs, and communicating information among planning, engineering, and maintenance teams. Due to the complex systems of pipes, structures, instruments, and rooms that are featured in such 3D scans, combinations of interaction features and interfaces are required for the best possible workflows. In this paper, we present the initial development of a Unity-based prototype system for visual analytics of pre-processed 3D scans utilizing a combination of widely used techniques like camera view bookmarks, point cloud class highlighting, and exploded views, together with modern Large Language Model (LLM) assisted visual analytics. The proposed prototype was demonstrated to two sewer utility companies, where prospective users provided a positive initial evaluation of the presented features, even with the added complexities of required pre-processing and an LLM model. The prototype will be used as a basis for the development of future systems. Prototype link - `https://github.com/Tannez/PointCloudInteractiveViewer/`

1. Introduction

Computing and analyzing point cloud datasets captured through the use of Laser Scanning or photogrammetry has become a cornerstone of research in infrastructure informatics and the management of the built environment (Tang et al., 2010, Poux and Billen, 2019). While these technologies have been used for broad landscape monitoring (Schütz, 2015), they are now critical for the high-fidelity documentation of complex industrial systems, including Right-of-Way (ROW) inspection of electricity power line networks (Yang et al., 2025a), and underground structures (Ma et al., 2025). To capture enough information and to represent the scanned surfaces, these point clouds contain millions or billions of points, making them hard to work with and requiring large storage space (Schütz et al., 2020). One of the biggest problems with working with large point clouds is the unintuitive and complex workflows for visualizing specific parts, gathering information, and understanding the overall structure of the captured objects. This happens especially for point clouds that contain hidden structures and multiple sub-objects, as understanding, visualizing, and later extracting important information run into both hardware and human understanding bottlenecks. This problem compounds when multiple people, from different backgrounds, need to analyze such point clouds, where precise notations and quick finding of information are needed.

There have been different ways for mitigating problems with visualization of large point clouds, like adaptive levels of detail (van Oosterom et al., 2022), injecting of metadata like classes and relations (Richter and Döllner, 2014), different rendering and detail accumulation (Futterlieb et al., 2016), and even using deep learning to regularize the unstructured point clouds (Zhang et al., 2022). These solutions alleviate many of the problems, but are directed towards professional use and do little for regular users who need to easily, at a glance, extract information from the point clouds.

In this paper, we present our work in developing a proof-of-concept solution for easier visualization of 3D-scan point clouds of sewer systems. We separate the proposed solutions into two primary modules. The first module focuses on introducing intuitive interaction techniques, such as camera view bookmarking, point cloud class filtering, and exploded views. This filtering allows users to toggle the visibility of specific classifications within the dataset to isolate relevant structures. The second module focuses on the utilization of Large Language Models (LLMs) as assistants to prospective users. This allows for interactions through a simple chat interface, lowering the barrier of entry to complex visualization tasks. The prototype was presented to and evaluated by representatives from two sewer utility companies and has been deemed useful. Users responded most favorably to the bookmarking and class highlighting features; while they remained more skeptical of the current LLM functionality, they expressed a clear interest in the introduction of further AI-driven features. An important point is that currently the solution assumes that the provided point clouds are pre-processed and not raw ones. Meaning that information about different sub-parts of the point cloud like rooms, objects, pipes, etc. has already been injected into the point cloud. Having said that this pre-processing can be done either manually or automatically by utilizing a deep learning point cloud segmentation model, and our proposed pipeline would work accordingly. A hypothesis is then specified - "**Does the introduction of advanced visualization and analysis features, and an LLM helper to a point cloud visualization pipeline enhance or hinder the working process of professional users**". From this hypothesis the two separate research questions can be derived - "*Does advanced visualization using bookmarking, semantic class highlighting and exploded views provide any benefits to professional users*" and "*Does the introduction of a chat-like LLM helper interface provide any benefits* *over the use of traditional UI for professional users*". The subsequent chapters will provide a better insight into these questions. The main contributions of this work can be summarized as:

- Developing a fully featured vertical slice prototype of an application for visualization and analysis of sewer system 3D-scan point clouds
- Introducing multiple visualization possibilities in the form of collaborative camera view bookmarking, semantic class highlighting/filtering, and exploded views of different parts of the point clouds
- Implementing an LLM virtual assistant that users can prompt with questions about the point cloud or ask it to provide specific visualizations directly.

2. Related Work

2.1 Software Solutions for Point Cloud Visualization

There exist many ready-made solutions for the visualization and rendering of large-scale point clouds. One widely used one is Potree (Schütz, 2023), a web-based renderer with a simple user interface for handling aspects such as point and distance measurements, clipping, and camera navigation. This simple user interface and workflow provided by Potree made it an inspiration for later developed tools such as PointView (IT34, n.d.), another solution created with the purpose of providing an intuitive workflow when exchanging and managing point clouds, 3D models, and detailed interactive maps. For more detailed use cases, tools such as CloudCompare (cloudcompare.org, 2009) and Meshlab (Cignoni et al., 2008) are used that provide additional features and advanced algorithms to accomplish tasks such as registration, resampling, color/normal/scalar fields handling, statistics computation, sensor management, interactive or automatic segmentation, display enhancement, etc. Finally, the SAMP Shared Reality visual workspace(SAMP, 2025) is an alternative solution that differs in that it allows for selecting specific instances directly within the point cloud, with a menu on the right showing their location within the hierarchy of instances. This visual workspace also has access to 2D sketches related to the data, allowing for direct comparison between the 2D sketches and the 3D visualization.

Together with analyzing these applications, we conducted discussions with 3D reconstruction professionals and tried to understand their needs and workflows. We identified several features that can be useful for the visualization of large, complex point clouds containing multiple sub-parts. One of the first useful features identified is the class selection and semantic segmentation visualization (Martinez-Rubi et al., 2015, Zhang et al., 2024). As many times smaller point cloud scans are combined into a larger one, we can utilize these given classifications of the points to visualize just a smaller subset of them with a specific color or highlight. Another feature is the exploded view, where different sub-parts of the 3D model can be moved dynamically to create a separation between them and give a better overview of complex structures (Bruckner and Groller, 2006, Gaarsdal et al., 2024). A third investigated navigational feature is the bookmarking of camera positions and orientations in the scene. This way, different views can be saved and shared between users for easier illustration of ideas. Similar features are described in Geographic Information System (GIS) applications like ArcGIS and ARCMap (ESRI, n.d.a, ESRI, n.d.b), as well as Building Information Modeling (BIM) viewers like Catenda BIM (Catenda, 2025).

2.2 Large Language Models and Point Cloud Understanding

Recent research has started looking into the interplay between LLMs and Point Clouds, primarily for comprehending 3D geometric and physical scenes (Xu et al., 2024, Ameen et al., 2025, Yang et al., 2025b). Runsen Xu et al. (Xu et al., 2024) have developed PointLLM, a multi-modal LLM (MLLM), capable of understanding colored point clouds of objects, perceiving object types, geometric, and appearance without concerns for ambiguous depth, occlusion, or viewpoint dependency. While experimental results demonstrated PointLLM's superiority over both 2D and 3D MLLMs on two separate datasets for various prompt types (ModelNet40 and Objaverse), the MLLM is only capable of understanding the structure and appearance of point clouds, but is not yet able to generate interactive 3D content or manipulate the point clouds in any way. Similar results can be seen in other studies (Ameen et al., 2025, Yang et al., 2025b), where LLMs can understand LiDAR-based data and provide information about it, but cannot affect the data.

3. Prototype Development

The prototype was developed in the Unity game engine, as it provides a robust and flexible basis for large-scale visualizations, and we can easily export the application for different operating systems, web, and mobile. For visualization of the point clouds, we use the open-source library BA_PointClouds (Fraiss, 2023), due to its extendability and ease of use, as well as the fact that it uses the same file format as industry standard applications like Potree. We additionally extend the library functionality to introduce the use of other point cloud attributes other than color and position. We add the possibility to visualize the class of each point as a color, as well as the intensity. The base library was also extended to construct and store sections of the point cloud in a tree-like hierarchy, making it easier to handle various class-specific manipulations that were developed, like the semantic class highlighting/filtering and the exploded view.

3.1 Interaction Options Development

Once a stable system was established for loading point clouds and reading their attributes, the next step was to develop a UI that can support complex 3D workflows, which is inspired by other widely used applications in the underground and sewer system 3D-scan visualization. We take inspiration from the positioning of options and granularity of interactions seen in Potree, as well as the simplicity of CloudCompare and PointView. An example of the viewer with a loaded point cloud and the interface can be seen in Figure 1.

Once these base development building blocks of the front and backend of the application were developed, we focused on introducing the visualization tools we found to be most widely used in the state of the art. For point cloud highlighting we use the class attributes that are present in many captured point clouds generated from multiple scans, representing different indoor rooms, parts of piping, halls, and machines. If these classes are not present from the capturing process, a preprocessing step for classification can be used, but this was beyond the scope of the development. These attributes were used

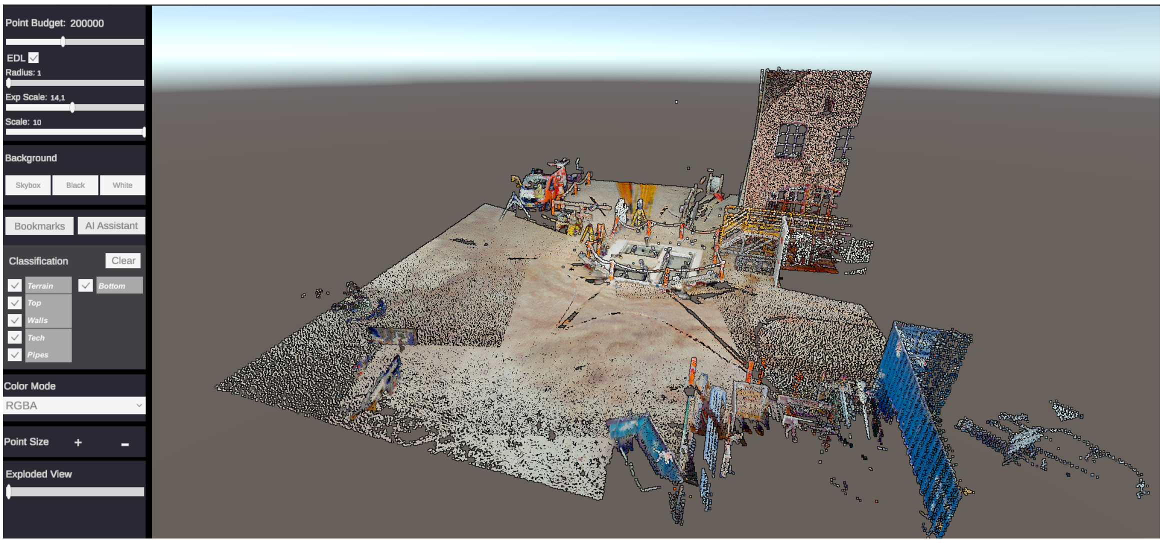


Figure 1. Example view from the application, showcasing the developed UI together with a loaded point cloud. Inspiration was taken from other software solutions like Potree and CloudCompare

to change the color of certain parts of the point clouds for easier visualization (Figure 2a). This way, a segmentation of the more complex point clouds can be achieved, focusing users' attention on specific sub-structures. Next, we introduce a bookmarking feature for camera positions. A user can move the camera to a specific point of the point cloud and click a button on the menu, creating a bookmark, saving how the camera was positioned and oriented. The camera view is also screenshotted and used as a thumbnail in visualizing the bookmark. These bookmarks are then saved at the top of the visual field, where, with one click, a user can go to the saved position (Figure 2b). Finally, we developed an exploded view of the different sub-parts of the loaded point cloud. Each sub-part of the larger point cloud after scanning is set up as a tree-like structure, with the full point cloud as the root, and the smaller objects and pipes are the leaves. This hierarchy is preset in many captured 3D reconstructions of point clouds for sewer systems, and each point of the point cloud is given this information, which is queried and used both for the segmentation selection and for the exploded view. The exploded view takes the sub-parts and moves their positions on the y-axis based on the child-parent structure. This way, smaller parts can be more easily viewed, and their hierarchy understood. The initial positions of the parts are saved, so the whole cloud can be as easily "un-exploded" (Figure 2c).

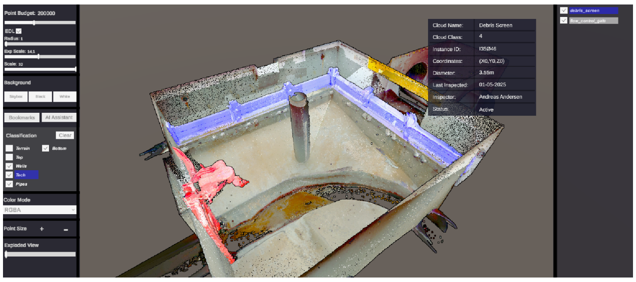
3.2 LLM Assistant Development

The idea of the LLM assistant was that it should do more than just give basic written information about the point cloud, as this is not enough incentive for long-term user interaction. The idea was thus expanded that the LLM can be used as both an information source and a replacement for the developed UI menu, where users can perform the same interactions with the point cloud but use natural language. This was done by specifically looking for keywords in the user input that would trigger a function response from the LLM. The basis for this was the library LLM for Unity by UndreamAI (undreamai, 2025). As a basis, we use the Llama 3.2 3B model with a context size of 5000 tokens, and 100% of the LLM to be used on the GPU at once.

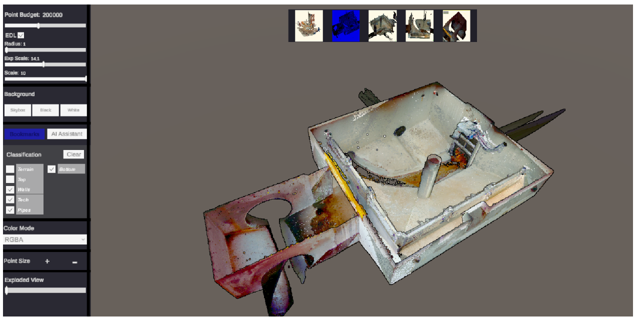
Through it, multiple agents can be spun up to handle different tasks. In our case, we create two agents: a Point Cloud Companion that provides information and insights into the point cloud but cannot change it, and a Function Handler agent that directly manipulates the point cloud and supports natural language interactions. First, the Point Cloud Companion agent picks up the message and provides feedback to the user through a chat UI that the request is being processed. It sends the message to the second agent. The Function Handler agent is prompted with a multi-choice function grammar setup that helps it distinguish if the user prompt contains a request for a specific functionality or a question. If the Function Handler does not detect a function, it switches control to the Point Cloud Companion, which answers the user. If a function is detected, then the Function Handler finds the best-matching function name and executes it. In both cases, an answer is sent back to the user. A flowchart of the pipeline is given in Figure 3, and a visual from the LLM assistant UI with a chat screen is shown in Figure 4. Finally, in Table 1, we show the two different prompts - one used when a function is detected as part of the user prompt and one when no function was detected - when it was just a conversation or question prompt.

Point Cloud Companion Prompt
"User Input: " + *message* + "." User Input is registered as Conversation and no function has been executed. (this next part is for you to understand what functions you have executed before to help better understand user intent. Please avoid mentioning this part to the user). Functions executed in scene: " + *functionContext*;
Function Handler prompt
"User Input: " + *message* + "." Input has made you do the following: " + *applyFunction.Item2* + ". Inform the user of your decision."

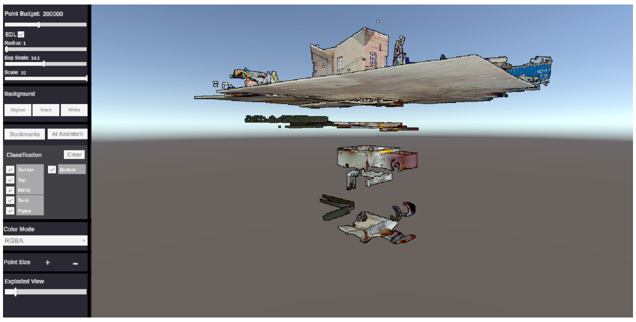
Table 1. Differences in the prompt string sent to the chat LLM based on whether user input is registered as a function or not. The *message* variable is the typed input sent by the user. The *function context* is a string that gets updated every time a function is called (essentially creating one long string of context). The *applyFunction.Item2* is a string set within the function handler script, based on the function being called.



(a) Point Cloud Selection and Segmentation



(b) Bookmarks of Views



(c) Exploded View of Sub-parts

Figure 2. Example of the different visualization functionalities developed for the prototype. First, a segmentation visualization that used the pre-defined parent-child connection in the provided point clouds and highlighted smaller objects (Figure 2a). Second, the bookmarked camera views that can directly put the camera at specific positions around and inside the point cloud (Figure 2b), and finally, the exploded view, which again used the parent-child connection between sub-point clouds to explode and visualize them separately (Figure 2c).

4. Prototype Presentation and Feedback

The developed prototype was demonstrated to two sewer utility companies and evaluated using expert interviews (Van Audenhove and Donders, 2019) and the think-aloud method (Wright and Monk, 1991), involving a total of 7 employees from the two companies. Their responsibilities ranged across project leaders, engineers, maintenance staff, and land surveyors, and they varied in their level of experience with 3D point cloud software, ranging from entry-level users who only had experience with basic viewing, to experts in 3D reconstruction. For each, a presentation was first given, followed by a demonstration of the prototype and free use. Participants were encouraged to discuss and give feedback throughout the time, and after the demonstration, an additional free-form discussion was conducted. The full process was audio-recorded after asking for consent from all participants.

This process was needed to gain knowledge about aspects such

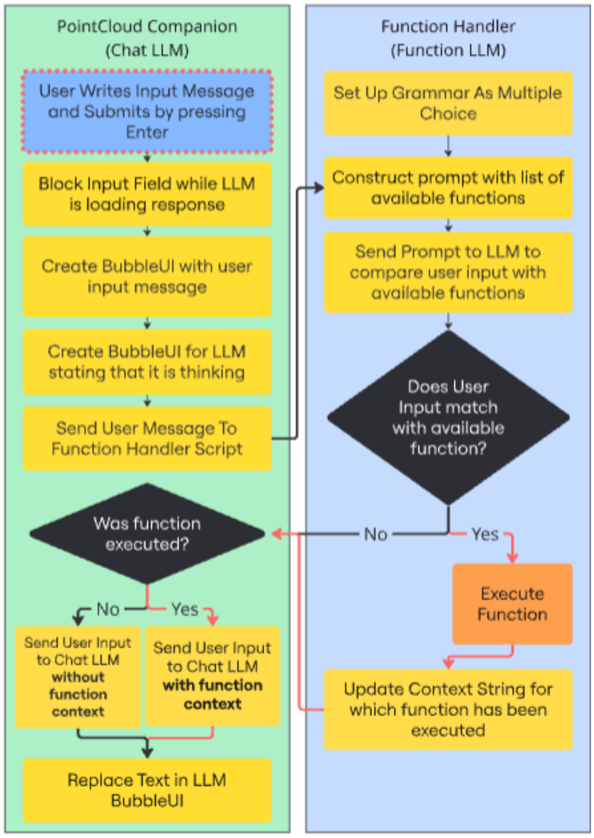


Figure 3. Flowchart of the LLM System. The message is first intercepted by the Point Cloud Companion agent, the user is informed through a chat UI, and the message is sent to the Function Handler agent. The agent then checks if the input matches any known function. If it does, it executes it and brings control back to the first agent. If there are no known function signatures, the Point Cloud Companion agent answers the user's query.

as processes, interactions, or routines employed when using point cloud visualization tools, and to discuss the usefulness of the presented features in the prototype. Both companies showed great interest in integrated features such as the bookmarks feature and segmentation cloud manipulation control, which both reviewed the different sub-structures of more complex point clouds. Primarily, the aspect of being taken directly to a given camera position within the 3D environment shows great promise, as users emphasize the struggle in having to use mouse scroll or dragging to reach areas of interest within the point cloud, leaving users confused or lost in the scene. The addition of an information window at specific instance locations adds to the usefulness, as not only are users guided by the program to relevant positions, but the desired information is presented in an easy and manageable manner when at these locations. Regarding the exploded view feature, the consensus among users was that it was seen as a gimmick-driven feature. While it could aid users in understanding component relationships, it did not provide insights into information that benefits the users in relation to their processes or routines.

As for the integrated LLM AI assistant, while initially impressed with the demonstrated capabilities, users discussed concerns for LLM precision when relaying information about point cloud metadata. The hallucinatory properties of LLMs (Ji et al., 2023, Oelschlager, 2024) present risks of reporting inac-

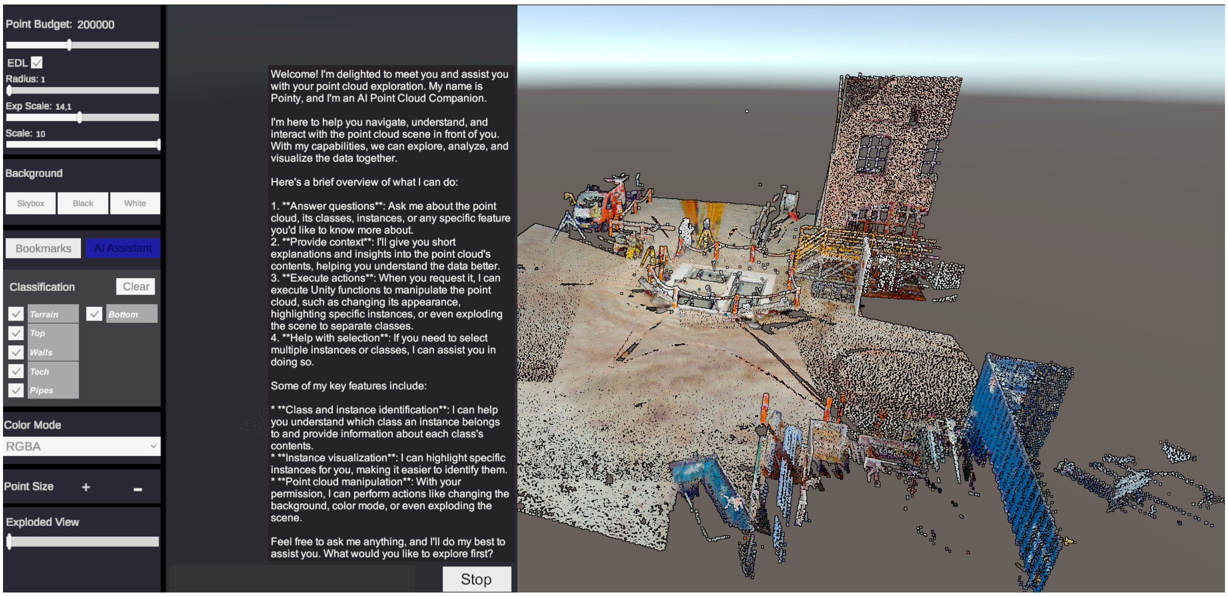


Figure 4. Example view from the LLM assistant, with its initial introduction and explanation of what it can do for the user.

curate measurements, which can lead to many problems within the industry. For this to be integrated, proper guard-railing of the LLM with specific insight into LLM decisions is therefore heavily required. However, an alternative integration is proposed by users in relation to the LLM being a learning tool, instead of something capable of direct point cloud manipulation. The LLM could provide links to learning material or highlight elements in the UI to aid users in learning how to find their desired information. This keeps the users in control without fully entrusting LLMs with the sensitive meta information. Other comments for the LLM specified the positive use case of it to ”remind” users how to use the program, as most of them are not frequent users and tend to forget how to extract important information: *”Close to 90% would find this relevant to help them (the LLM) [...] most of the people who actually use this, only use it very few times. I would say once per quarter. Then we are a few, I would say under 10% who use it weekly.”*. Finally, users pointed out the possibility for the LLM to extract hidden or hard-to-extract metadata information from the point cloud: *”The more meta information you can stuff into such a point cloud, the more interesting such a chat function will become [...] because, there can exist many meta informations hidden somewhere deep, that is incredibly interesting, but that a regular user would not be able to find, but could find by asking, and then it finds it for them.”*. The consensus was cautiously optimistic, with many hesitant to trust the LLM without a human-in-the-loop checking its work: *”I do not trust LLMs in relation to their precision in delivering data. If it should be able to deliver data, then it must have a 'show-your-work' function, so that you know why it got the result that it got.”*

5. Discussion

While the prototypes allowed for exploration of new features, a limited set of new ones was actually explored. In total, the new features created were the class and instance marking, bookmarks, exploded view, LLM interface, and, later, the addition of the instance information window. More features within the world of 3D interaction could therefore still be explored to see if users might find them more beneficial than the features presented in this paper. An example of a feature that was considered but ultimately scrapped was an X-ray flashlight visual that would let users see through layers of the point cloud and focus on smaller internal objects like pipes, wires, levers, and terminals. This feature was scrapped as it broke the proposed workflow, but it was later discussed as part of the feedback from the sewer utility companies. The three proposed interaction techniques were seen as useful and potentially workflow-changing by the companies, as none of the state-of-the-art applications contained all of them and gave users the possibility to use them interconnectedly. The prototype also works with already pre-processed point clouds that already contained hierarchies of sub-point clouds. In the case of using raw data, a pre-processing step will need to be done, either manually by a person, or automatically using a deep learning model for segmenting the different sub-parts.

The LLM-power helper was seen as the most polarizing feature, as it was viewed with the most skepticism from long-time professionals, but was also discussed as the one feature that could most easily bridge the gap between newcomers and veterans. Giving a simple text-based interface to less technical users to present complex views and communicate ideas was seen as a way to modernize workflows. To mitigate the possibilities of hallucinations and incorrect workflows, a tighter coupling between the data and LLM should be established, where the LLM has direct access to the point cloud data, even outside of the scene. An additional step-by-step protocol should be established where the LLM can be forced to do the required processing and analysis in smaller increments, which the user can evaluate and agree with or return for additional changes.

Finally, to truely understand if the proposed workflows help users, a larger scale user study needs to be conducted, where users are given a task to complete using either only the simple interface, or the enhanced functionality of instance marking ,boomarks and exploded views, or finally the LLM interface. The amount of errors, time-for-completion and a think-aloud method can be employed to better capture the benefits and drawbacks of the proposed system.

6. Conclusions

We presented a vertical-slice prototype for visualizing and analyzing large point clouds from 3D scans of sewer and water treatment facilities. Inspired by existing tools, it integrates established techniques such as segmentation highlighting, camera view bookmarking, and exploded views, and augments them with an agent-based LLM interface that complements or replaces menu-driven interactions. The prototype was demonstrated to stakeholders from sewer utility companies. Bookmarking and segmentation were particularly well received, while reactions to the exploded view were mixed. The LLM component was met with some skepticism regarding data reliability, but was also appreciated as an educational aid and a quick reorientation tool.

Future work will extend the prototype with improved data visualization features, including segmentation planes, enhanced sub-object isolation, and additional shading options. The LLM will be further developed as an on-demand assistant, accessible via contextual menus and systematically evaluated in practice.

Acknowledgements

This work was funded by VUDP, a Danish network working to improve the efficiency and quality of the Danish water sector. We would also like to thank our Danish partner companies for their contributions to this project: Aarhus Vand A/S, Novafos A/S (sewer utility), and LE34 A/S (land surveying).

References

Ameen, H., Núñez-Seoane, A., Conde, D., Soilán, M., 2025. Integrating Multimodal Large Language Models with Point Clouds for Urban Vehicle Detection. *Inzynieria Mineralna-Journal of the Polish Mineral Engineering Society*, 2.

Bruckner, S., Groller, M. E., 2006. Exploded views for volume data. *IEEE transactions on visualization and computer graphics*, 12(5), 1077–1084. doi.org/10.1109/TVCG.2006.140.

Catenda, 2025. Canteda bim. (10 December 2025).

Cignoni, P., Corsini, M., Ranzuglia, G., 2008. MeshLab: an Open-Source 3D Mesh Processing System. *ERCIM News*, 2008(73), 129–136.

cloudcompare.org, 2009. Cloudcompare. (10 December 2015).

ESRI, n.d.a. Arcgis: Esri's comprehensive geospatial platform. (10 December 2025).

ESRI, n.d.b. Arcmap. (10 December 2025).

Fraiss, S. M., 2023. Ba_pointcloud. (15 December 2025).

Futterlieb, J., Teutsch, C., Berndt, D., 2016. Smooth visualization of large point clouds. *IADIS International Journal on Computer Science and Information Systems*, 11(2), 146–158. http://publica.fraunhofer.de/dokumente/N-453338.html. (15 December 2025).

Gaarsdal, J., Haurum, J. B., Wolff, S., Madsen, C. B., 2024. Assemblynet: A point cloud dataset and benchmark for predicting part directions in an exploded layout. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 5836–5845.

IT34, n.d. Pointview®. a world of data at your fingertips. (10 December 2015).

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., Fung, P., 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12), 1–38. doi.org/10.1145/3571730.

Ma, Y., Wang, S., Xin, G., Li, B., Fang, H., Lei, J., Du, X., Wang, N., Di, D., 2025. A state-of-the-art-review of underground concrete sewage pipelines detection technologies. *Measurement*, 242, 116268. doi.org/10.1016/j.measurement.2024.116268.

Martinez-Rubi, O., Verhoeven, S., Van Meersbergen, M., Schütz, M., Van Oosterom, P., Gonçalves, R., Tijssen, T., 2015. Taming the beast: Free and open-source massive point cloud web visualization. *Capturing Reality Forum*, 2016.

Oelschlager, R., 2024. Evaluating the impact of hallucinations on user trust and satisfaction in llm-based systems.

Poux, F., Billen, R., 2019. Voxel-based 3D point cloud semantic segmentation: Unsupervised geometric and relationship featuring vs deep learning methods. *ISPRS International Journal of Geo-Information*, 8(5), 213. doi.org/10.3390/ijgi8050213.

Richter, R., Döllner, J., 2014. Concepts and techniques for integration, analysis and visualization of massive 3D point clouds. *Computers, Environment and Urban Systems*, 45, 114–124. doi.org/10.1016/j.compenvurbsys.2013.07.004.

SAMP, 2025. Your ai-generated visual workspace. (10 December 2025).

Schütz, M., 2015. Potree: Rendering large point clouds in web browsers. PhD thesis, Technische Universität Wien.

Schütz, M., 2023. Potree: Webgl point cloud renderer for large point clouds. GitHub repository. (15 December 2025).

Schütz, M., Ohrhallinger, S., Wimmer, M., 2020. Fast out-of-core octree generation for massive point clouds. *Computer Graphics Forum*, 39number 7, Wiley Online Library, 155–167.

Tang, P., Huber, D., Akinci, B., Lipman, R., Lytle, A., 2010. Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques. *Automation in construction*, 19(7), 829–843. doi.org/10.1016/j.autcon.2010.06.007.

undreamai, 2025. Llmunity. (9 December 2025).

Van Audenhove, L., Donders, K., 2019. Talking to people iii: Expert interviews and elite interviews. *The Palgrave handbook of methods for media policy research*, Springer, 179–197.

van Oosterom, P., van Oosterom, S., Liu, H., Thompson, R., Meijers, M., Verbree, E., 2022. Organizing and visualizing point clouds with continuous levels of detail. *ISPRS Journal of Photogrammetry and Remote Sensing*, 194, 119–131. doi.org/10.1016/j.isprsjprs.2022.10.004.

Wright, P. C., Monk, A. F., 1991. The use of think-aloud evaluation methods in design. *ACM SIGCHI Bulletin*, 23(1), 55–57. doi.org/10.1145/122672.122685.

Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D., 2024. Pointllm: Empowering large language models to understand point clouds. *European Conference on Computer Vision*, Springer, 131–147.

Yang, B., Dong, Z., Liang, F., Mi, X., 2025a. *Ubiquitous Point Cloud: Theory, Model, and Applications*. 1, 1 edn, CRC Press.

Yang, S., Liu, J., Zhang, R., Pan, M., Guo, Z., Li, X., Chen, Z., Gao, P., Li, H., Guo, Y., Zhang, S., 2025b. Lidar-llm: Exploring the potential of large language models for 3d lidar understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39number 9, 9247–9255.

Zhang, Q., Hou, J., Qian, Y., Chan, A. B., Zhang, J., He, Y., 2022. Reggeonet: Learning regular representations for large-scale 3d point clouds. *International Journal of Computer Vision*, 130(12), 3100–3122. doi.org/10.1007/s11263-022-01682-w.

Zhang, Z., Ding, J., Jiang, L., Dai, D., Xia, G., 2024. Freepoint: Unsupervised point cloud instance segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 28254–28263.