"Colorful Smoke" Visual Metaphor for Temporal Occupancy in Urban Digital Twins

Olga Shkedova [1]*, Jeson Lonappan[1], Monika Sester[1]

[1] Institute of Cartography and Geoinformatics, Leibniz University, Hannover, Germany-
Olga.Shkedova@ikg.uni-hannover.de, Jeson.Lonappan@ikg.uni-hannover.de, Monika.Sester@ikg.uni-hannover.de

Keywords: Visual Metaphor, Urban Digital Twin, Occupancy visualization, 3D visualization, voxel representation.

Abstract

Urban digital twins (UDTs) play a pivotal role in advancing smart city development, enabling virtual representations of complex urban environments and supporting advanced planning, monitoring, and decision-making. Despite significant progress, challenges remain in intuitively visualizing dynamic, heterogeneous data, managing real-time updates, and representing temporal occupancy in a comprehensible way. This research introduces a visual metaphor, "Colorful Smoke", for temporal space occupancy visualization within voxel-based UDTs. The approach first generates a voxel-based urban digital twin by fusing CityGML, Digital Terrain Model (DTM), and point cloud data. Elements assumed to be static at the time of observation are represented by voxels, whereas dynamic objects, such as vehicles and pedestrians, are represented using the 'Colorful Smoke' metaphor, in which smoke density encodes occupied space, transparency indicates the probability of occupancy, and color represents the type of dynamic object. In contrast to existing voxel-based methods, this design enables users to intuitively perceive temporal occupancy, assess data reliability, and identify areas of change in urban spaces without visual overload.
Future work will enhance the metaphor's clarity through additional colors and adjustment of rendering parameters, and evaluate its perceptual effectiveness via user studies and continuous dynamic updates.
The proposed "Colorful Smoke" approach offers a scalable and intuitive method for representing temporal occupancy in UDTs, bridging the gap between complex data and human perception, and providing a foundation for future dynamic urban visual analytics.

1. Introduction

Urban digital twins (UDTs) have emerged as a key technology in smart city development, enabling virtual representations of complex urban environments. One of the main characteristics of digital twins is the data synchronization mechanism between the virtual and physical worlds (Fuller et al., 2020). Several UDT applications are currently in operation. The Fishermans Bend Digital Twin provides 2D/3D visualization, integrates heterogeneous formats and real-time transport data, and supports collaborative planning (Chehrehbargh, 2022). Zurich's 3D city model enables environmental monitoring, energy assessment, and urban planning analyses (Schrotter and Hürzeler, 2020). Rotterdam 3D combines open 3D data with real-time traffic, parking, air quality, and waste-management information (City of Rotterdam, 2025; Coumans, 2019).

Despite the significant development of modern UTDs, a number of challenges still persist. Ferré-Bigorra et al. (2022) analysed 131 studies on UDTs and emphasized the need for more intuitive methods to represent complex, heterogeneous datasets, address data quality issues, and overcome real-time analysis bottlenecks. Jeddoub et al. (2023) analysed 22 operational UDTs and identified key technical challenges, including data heterogeneity and automated real–virtual synchronization. Martella et al. (2023) reviewed state-of-the-art UDT platforms, highlighting their role in supporting urban planning and development through advanced visualization, data analysis, and simulation tools for evaluating scenarios and interventions. The study also identified 3D visualization as a core component of UDT platforms, allowing users to intuitively explore and interact with urban environments. While 3D visualization plays a crucial role in urban digital twins, most modern UDTs prioritize photorealistic representations and the use of color as a visual variable for semantics, whereas the development of application-specific visualization approaches remains limited compared to the wide range of potential uses (Dembski et al., 2020; City of Helsinki, 2025; Lin et al., 2024; Boccardo et al., 2024).

To address these concerns, an alternative to the CityGML and BIM standards for 3D city models was proposed in our joint work Mortazavi et al. (2023), enabling real-time updates of dynamic UDTs and providing a uniform structure for heterogeneous datasets. This method decomposes the 3D urban environment into a unified voxel grid that stores spatial, temporal, and semantic information, allowing each voxel to be updated independently. Furthermore, semantic visualization in voxel-based UDTs was proposed in our subsequent work Shkedova and Sester (2024), using visual variables such as transparency, size, and color, along with animations, to convey space occupancy, an approach significant for modern space management applications. Although these researches show considerable potential, several limitations remain. Voxel-based UDTs in these studies were generated from precise point clouds, but acquiring data for large urban areas, including rooftops, is time-consuming. Visualizing semantic attributes in small voxels is challenging, and animations, while useful for static analysis, are less effective in dynamic scenarios where occupancy changes over time, causing voxels to appear and disappear in the digital urban model.

In response to these challenges, this research makes two contributions. The first, with a primary focus, is a 3D visualization approach for temporal occupancy probability, representing voxels of dynamic objects using a "Colorful Smoke" metaphor to simplify semantic representations of temporal data, reduce visual overload from small voxels, and intuitively convey the likelihood of temporary space occupancy. The second is a data aggregation method for voxel-based UDT, modeling of urban scene by fusing City Geography Markup Language (CityGML), Digital Terrain Model (DTM), and point cloud data.

* Corresponding author

2. "Colorful Smoke" concept

The proposed visual metaphor represents the temporal occupancy of space by dynamic objects within a UDT, effectively distinguishing them from the static scene and conveying occupied areas based on occupancy probabilities derived from sensor data.

Smoke, like fog or clouds, is perceived as inherently temporal due to its tendency to disperse. We leverage this intuitive association to depict dynamic objects that occupy space temporarily and may disappear in the next UDT update, reflecting real-world behavior. This metaphor applies to cars, bicycles, pedestrians, and other transient entities. Smoke characteristics such as density and opacity indicate the likelihood of space occupancy, while color conveys the object type. Thus, a single visual metaphor represents three aspects: temporal occupancy, occupancy probability, and object type.

3. Related work

A visual metaphor is a way of communicating an idea by representing it through imagery associated with a different, often more familiar, concept. It works by transferring recognizable attributes from one domain (the source) to another (the target), so that the viewer understands the intended meaning through analogy rather than direct explanation (Forceville, 1996).

Wijayawardena et al. (2023) reviewed 3D metaphoric information visualization and identified natural phenomena as a distinct category of metaphors for 3D visualizations. For example, Würfel et al. (2015) visualized software metric trends on a static map using rain and fire, Teles et al. (2020) used fog to represent 3D air quality data and Vetter (2023) applied fog to indicate non-penetrable objects in 3D worlds. Somanath et al. (2024) used wind effect, created by 2 million animated particles, to show NO_2 concentration. Zhang et al. (2011) simulated weather phenomena such as fog, snow, and rain, in combination with scene illumination, to enhance realism, immersion, and interactivity in a virtual environment.

Natural phenomena in UDTs are commonly simulated using various platforms and computer graphics techniques. These simulations typically represent realistic phenomena, such as fire, floods, atmospheric clouds, and smoke, to support emergency scenario modeling rather than convey abstract or metaphorical meanings (Rauer-Zechmeister et al., 2024; Kumar et al., 2018; Herman et al., 2017; Vanella et al., 2021; Steinicke et al., 2008; Zamri and Sunar, 2014). For the creation of "Colorful Smoke," we employed standard techniques for simulating and visualizing natural phenomena, such as ray marching (Levoy, 1988; Hart, 1996; Zhou et al., 2008; Bass and Anderson, 2013; Ren and Nakata, 2024). Ray marching determines pixel color and opacity by sampling along rays through a 3D volumetric dataset, such as smoke. Unlike surface-based rendering, the rays accumulate contributions from the volume - integrating emission, absorption, and scattering, enabling realistic visualization of semi-transparent or fully volumetric phenomena. Further details are provided in Section 4.4.4

4. Methodology

To implement the proposed visual approach, the urban digital environment must first be generated. A voxel-based representation of the urban digital twin is adopted as the foundation due to its suitability for temporal updates and discrete spatial structure, which supports 3D "Colorful Smoke" modeling. Since the study focuses on temporal occupancy visualization, it is assumed that static and dynamic objects occupy the urban space at the observation time.

Figure 1 represents the main steps of the workflow for the voxel-based UTD with the proposed visual-metaphor implementation. Each step will be described in detail in the following sections.

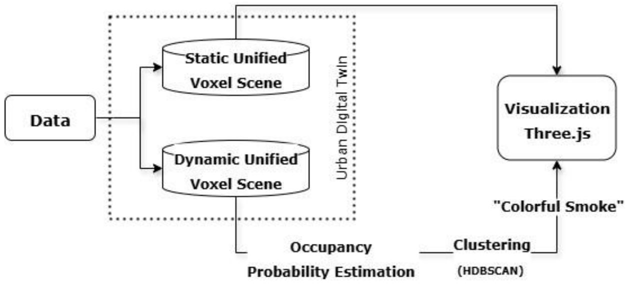

Figure 1. Workflow of the voxel-based UTD with "Colorful Smoke".

4.1 Data

Considering that the data must represent both static and dynamic objects, a busy street intersection in Hannover is selected for the experiment. The area is surveyed using the Riegl VMX-250 Mobile Mapping System (MMS) to obtain a high-accuracy point cloud with registered dynamic objects. To address the spatial limitations of the point cloud, additional data sources, including CityGML and DTM, are incorporated to complete the urban environment with representations of buildings and terrain.

4.1.1 Point Cloud: The obtained point cloud is composed of 23,988,182 points. For object recognition, the scene is classified using the Kernel Point Convolution (KPConv) approach (Hugues, 2019), resulting in 12 object classes: building, road, fence, streetlight, traffic sign, treetop, tree trunk, bush, terrain, person, car, and bike. Figure 2 shows the classified point cloud of the street intersection.

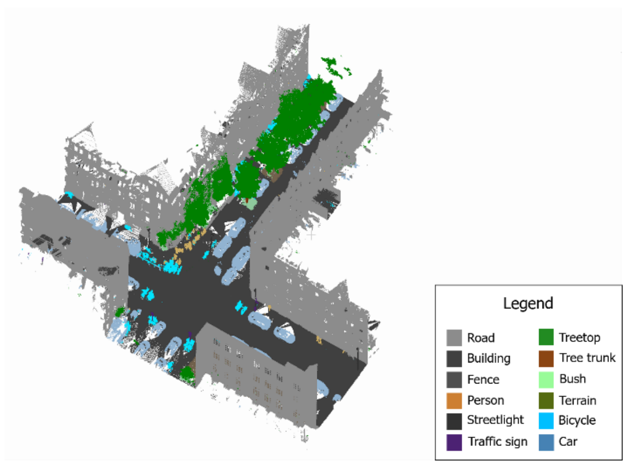

Figure 2. Classified point cloud.

The classification procedure enables the separation of points corresponding to potentially dynamic objects, which temporarily occupy space, from those assumed to represent permanent occupancy. Accordingly, points classified as "person," "car," and "bike" define the dynamic scene, while all other points represent the static scene.

As shown in Figure 2, the MMS captured building facades in great detail but was unable to capture entire structures, such as walls and roofs. The point cloud also exhibits gaps in occluded areas, for example, behind parked cars, which is a common limitation of LiDAR-based mobile mapping. To address these issues, we enhance the completeness of the static scene by fusing CityGML 3D building models with points classified as "building" and the DTM with points classified as "road" and "terrain" within a regular voxel grid. A detailed description of this procedure is provided in Section 4.2. The dynamic scene, in turn, is the focus of this research, as the "Colorful Smoke" visual variable is designed to visualize temporal space occupancy by dynamic objects. An explanation of the dynamic voxel scene processing is provided in Section 4.3.

4.1.2 CityGML: The CityGML data were obtained from the OpenGeoData portal of Lower Saxony (Niedersachsen, 2019). The 3D building models on this platform are derived from Germany's official digital cadastre system (ALKIS) building ground plans, DTM, and 3D measurement data (Niedersachsen Landesvermessungsamt, 2019). For this research, LoD2 building models are used. The positional accuracy of LoD2 corresponds to the underlying building floor plans, and the height accuracy is approximately 1 m. The buildings are modeled using standardized roof shapes, such as gable or hip roofs. The CityGML LoD2 tile fragment used in this research is shown in Figure 3 (left).

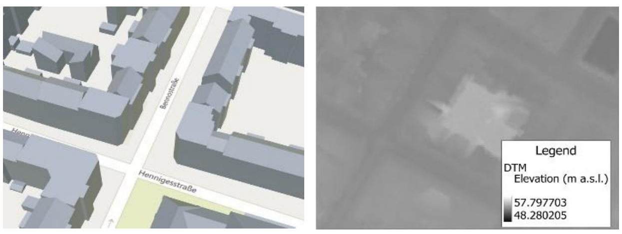

Figure 3. From left to right: CityGML tile fragment, DTM tile fragment.

4.1.3 DTM: The DTM was retrieved from the same source as the CityGML data. It was generated from laser scan point clouds acquired via Airborne Laser Scanning (ALS) and provides a geometric resolution of at least 4 points per m². The grid spacing is 1 m (DTM1). The DTM tile fragment of the experimental area is shown in Figure 3 (right).

4.2 Static Scene Generation

For the static voxel scene generation, a systematic multi-source data integration framework using heterogeneous geospatial datasets is developed. The methodology employs three sequential semi-automated pipelines within a PostgreSQL/PostGIS spatial database environment enhanced by the 3DCityDB schema (PostGIS, 2025; PostgreSQL, 2025; Yao et al., 2018).

Initially, the "building" point cloud is integrated with CityGML LoD2 building models using a thorough alignment process that includes criteria to identify regions with sufficient data overlap, followed by applying buffer zones and clipping around the chosen building facades (Wysocki et al., 2021). Spatial registration is achieved through multi-stage alignment with height correction using CityGML absolute height attributes, coarse alignment via RANSAC plane fitting to distinguish modeled walls rom unmodeled architectural features (Fischler and Bolles, 1981), and fine alignment using the Iterative Closest Point (ICP) algorithm, resulting in improved positional accuracy (Besl and McKay, 1992). Subsequently, the DTM is refined by integrating points classified as "road" and "terrain" through custom tuned Cloth Simulation Filtering (CSF) (Zhang et al., 2011) and Constrained Delaunay Triangulation (CDT) with building footprints serving as breaklines to rectify elevation inconsistencies at terrain-building intersections (Chew, 1987).

The final voxelization stage establishes a unified discrete spatial grid through systematic volumetric decomposition. CityGML is voxelized in the database using PostGIS functions to preserve complex semantic attributes, while the DTM and point cloud datasets are voxelized by directly mapping their spatial coordinates to the unified voxel grid of 10 cm size. The CityGML voxels are assigned corresponding semantic identifiers, enabling semantic queries and topology preservation. For DTM and point cloud data, the voxelization process allocates each point to its respective voxel index based on spatial location.

All three datasets are then integrated within the 10 cm voxel grid. Terrain voxels from the DTM form the base layer. CityGML voxels supersede the terrain wherever buildings are present, thereby embedding semantic richness. The voxels derived from the point cloud subsequently provide further refinement to building geometry and attributes wherever higher detail and density are available. This systematic approach combines the advantages of each data source by utilizing the consistent terrain representation from the DTM, the rich semantic information from CityGML, and the detailed geometric accuracy provided by the MMS data. As a result, both semantic and geometric information are preserved throughout the unified voxel domain. The described framework is schematically represented in Figure 4.

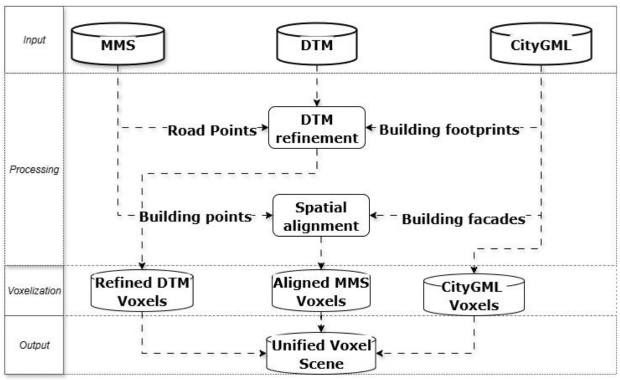

Figure 4. The static voxel scene generation framework.

4.3 Dynamic Scene Generation

To define the space temporarily occupied by dynamic objects, the point cloud of the dynamic scene is first voxelized within a unified grid of 10 cm, following a procedure similar to that used for the static scene. Figure 5 shows the voxelized dynamic objects represented by the voxel point centers.

4.3.1 Occupancy Probability Estimation: The visual metaphor aims not only to indicate which space is occupied but also to convey the likelihood of occupancy, enabling a more precise and continuous visual analysis of temporal occupancy based on spatially limited sensor data. After voxelization, each voxel contains a variable number of points determined by local point density, which is influenced by scanning geometry (e.g., occlusions, incidence angles, sensor distance) and urban environment characteristics (e.g., facades, vegetation, open

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W19-2025-127-2026 | © Author(s) 2026. CC BY 4.0 License.

ground). Consequently, the point count within each voxel can serve as a measure of occupancy probability for the metaphor. Accordingly, the occupancy probability p of each voxel is computed as follows:

$$p_i = \frac{c_i}{c_{max}}, \tag{1}$$

where c_i = point count in the i-th voxel
c_{max} = maximum point count across all voxels
p_i= = normalized occupancy probability of i-th voxel

This results in probability values ranging from 0 to 1. The dynamic scene voxels are represented by their point centers, colorized according to the voxel probability values in Figure 6. As can be observed, the highest occupancy probabilities are associated with voxels that were closer to the MMS during the scanning process, not occluded, and had surfaces oriented perpendicular to the laser beam.

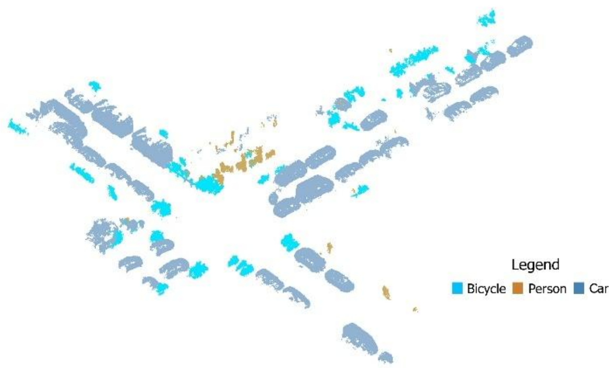

Figure 5. Dynamic voxel scene.

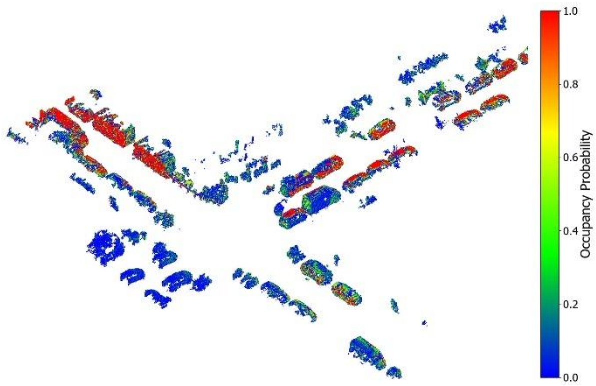

Figure 6. Occupancy probability of the dynamic voxel scene.

4.4 "Colorful Smoke" Visual Metaphor Implementation

Following the concept described in Section 2, temporally occupied space is visualized as smoke, with density, color intensity, and opacity co-varying. Voxels with higher occupancy probability are depicted as denser, more opaque smoke with stronger color intensity, whereas voxels with lower probability appear as sparser, more transparent smoke with reduced color intensity. To implement this visualization, a GPU-based rendering pipeline is developed using the Three.js library (Cabello and contributors, 2010–2025) for a scene management and GLSL shaders (Khronos Group, 2024) for high-performance computation. The method generates a 3D scalar field representing per-voxel probabilities, which is rendered using the ray marching technique with trilinear interpolation (Bourke, 1999). The main steps of the pipeline are illustrated in Figure 7. Each of these steps is briefly outlined in the following sections of this chapter.

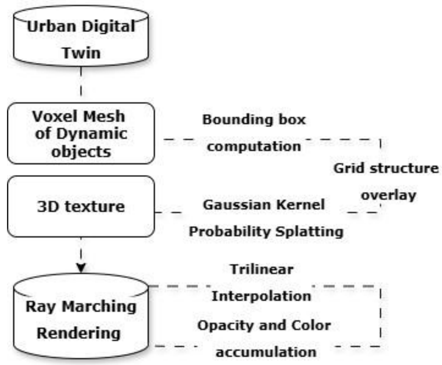

Figure 7. "Colorful smoke" visualization pipeline.

4.4.1 Urban Digital Twin: The voxel-based UDT is generated by integrating dynamic and static scenes, described in Sections 4.2 and 4.3, into a uniform grid, where each voxel stores spatial data (x, y, z coordinates) and attribute data, including color, class, CityGML building attributes, and probability values for dynamic objects.

4.4.2 Voxel mesh of dynamic objects: For simplicity in controlling and analyzing the visual metaphor, dynamic objects are clustered using the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm (Campello et al., 2013). Consequently, each cluster corresponds to a single dynamic object or a small group of dynamic objects within the voxel instance mesh, where each voxel stores a probability value.

4.4.3 3D texture: The world-space positions of all instances in a mesh are transformed and aggregated to compute a global axis-aligned bounding box. This bounding box defines the extents of a uniform 3D voxel grid with resolution (N_x, N_y, N_z), where the voxel size is selected according to the desired sampling density - 10 cm in our case. Next, to convert discrete instance probabilities into a continuous volumetric probability field, for each instance probability mass is distributed to surrounding voxels according to a discrete isotropic Gaussian kernel (Westover, 1991):

$$w(\Delta v) = \exp\left(\frac{\Delta v_{x^2} + \Delta v_{y^2} + \Delta v_{z^2}}{2_{\sigma^2}}\right), \tag{2}$$

where v = (v_x, v_y, v_z) = voxel coordinate in the grid
Δv = voxels offset from the voxel center
σ = smoothing parameter

This step produces a continuous scalar field while preserving local maxima at instance centers. It helps reduce noise in the data and creates smooth gradients between occupied and unoccupied space. However, to maintain accurate occupancy boundaries, the value of σ should be kept relatively low, as high values can excessively blur the probability distribution and reduce spatial precision. Subsequently, the voxel grid, defined by the bounding box and resolution, is populated with probability densities obtained using a Gaussian kernel (Equation 2). This data is then stored in a floating-point RGBA 3D texture, where the RGB channels encode the base color, and the alpha channel holds the accumulated probability density.

4.4.4 Ray Marching Rendering: Rendering is performed in the fragment shader by ray marching through the volumetric data obtained for each cluster. For every pixel, a camera ray r is cast as follows:

$$r(t) = o + td, \quad (3)$$

where o = camera origin
d = normalized ray direction
$t \geq 0$ = distance along the ray from the origin

The ray is intersected with the 3D texture volume box, yielding entry and exit parameters t_{enter}, t_{exit}, which are sampled at fixed step size s within $[t_{enter}, t_{exit}]$. At each marching step t_i, the position P_i along the ray is normalized into local texture coordinates $\mathrm{f}_i \in [0,1]^3$:

$$\mathrm{f}_i = \frac{P_i - b_{min}}{b_{max} - b_{min}}, \quad (4)$$

where $P_i = r(t_i)$ = actual 3D position at step i
b_{min} = minimum bound of the volume box
b_{max} = maximum bound of the volume box

Figure 8 illustrates the ray marching procedure through the volume box (N_x, N_y, N_z).

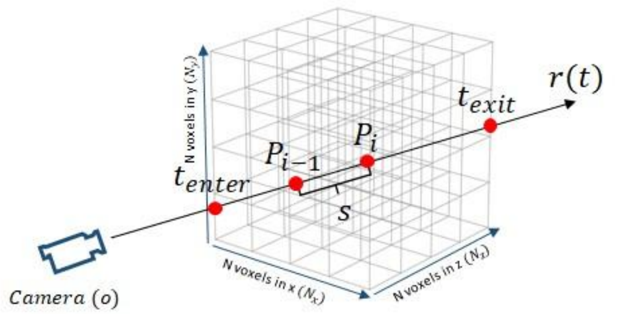

Figure 8. Ray marching procedure through the volume box.

Additionally, to perform smooth sampling between grid points in all directions and ensure a visually continuous and natural, smoke-like appearance of the metaphor without introducing extra blur, trilinear interpolation is applied. Therefore, at each marching step i the texture coordinate $\mathrm{f}_i \in [0,1]^3$ is converted to continuous voxel coordinates as follows:

$$\mathrm{v}_i = \mathrm{f}_i \cdot (N_x, N_y, N_z) = (u, v, w) + (f_x, f_y, f_z), \quad (5)$$

where $(u, v, w) \in Z^3$ = integer voxel corner indices
$(f_x, f_y, f_z) \in [0,1]^3$ = fractional cell offsets

The scalar field value at f_i is computed by trilinear interpolation of the voxel values $c_{i,j,k}$ at the eight corners of the cell, with $i, j, k \in [0,1]$, as:

$$\begin{aligned} C(\mathrm{f}_i) = \; & mix(mix(mix(c_{000}, c_{100}, f_x), \\ & mix(c_{010}, c_{110}, f_x), f_y), \\ & mix(mix(c_{001}, c_{101}, f_x), \\ & mix(c_{011}, c_{111}, f_x), f_y), f_z), \end{aligned} \quad (6)$$

where $mix(a, b, t) = a(1-t) + bt$ = linear function
$C(\mathrm{f}_i)$ = smooth scalar sample along the ray

Furthermore, to simulate the light absorption and opacity contribution of the volumetric probability density sampled along the ray, the based opacity α is computed as follows:

$$\alpha_i = 1 - \exp(-C(\mathrm{f}_i)\, sO), \quad (7)$$

where $C(\mathrm{f}_i)$ = interpolated value at the current sample
s = fixed sampling step size along the ray
O = scaling factor that controls opacity sensitivity

Finally, color and opacity are assembled to progressively build the final pixel color while handling transparency and occlusion.

5. Experiments and Results

This section examines the impact of processing parameters, including the scaling factor O and sampling step s, on smoke visualization, and presents the UTD with static voxels and temporarily occupied space depicted using the “Colorful Smoke” metaphor.

A comparative view of the “Colorful Smoke” visualization for different values of the scaling factor O and sampling step s is presented in Figure 9. For visual analysis, example clusters corresponding to three dynamic object classes (“bicycle” - here, a group of bicycles, “person”, and “car”) are selected. The first experiment investigates the influence of O on opacity sensitivity, with values ranging from 1 to 5, 10, 20, and 30.

Figure 9 shows that when O = 1, the opacity is very low, and the clusters appear as barely perceptible, almost colorless smoke. As O increases, colors become more pronounced, particularly in regions with high-probability voxels or overlapping low-probability voxels. The smoke also becomes denser, with areas that previously appeared as semi-transparent grey volumes transitioning into more vividly colored regions. This effect is especially noticeable at the corner of the car’s hood and bumper. Since the 3D smoke effect is observed in a 2D image, red solid circles were added to highlight areas with minimal voxel overlap, where changes in the visual variable are most apparent, and dashed red circles were used to highlight an example of low-probability voxel overlap. In the bicycle instance, the high-probability region expands as O increases, evolving from a small, shadow-like spot at O = 1 to a large, vividly colored area surrounded by a thin shade at O = 30. Additionally, a low-probability region becomes visible starting at O = 10, appearing to the right of the highlighted area due to the overall increase in opacity. A similar trend is observed for the “person” and “car” clusters: as O increases, smoke density grows and object shapes become more clearly defined. Additionally, a striped effect is visible in the car example (particularly on the hood), which becomes more prominent at higher O values (20 and 30) and highlights noise in the data. It is important to note that for the visual metaphor, a Gaussian kernel was applied according to Equation 2 with σ = 0.4.

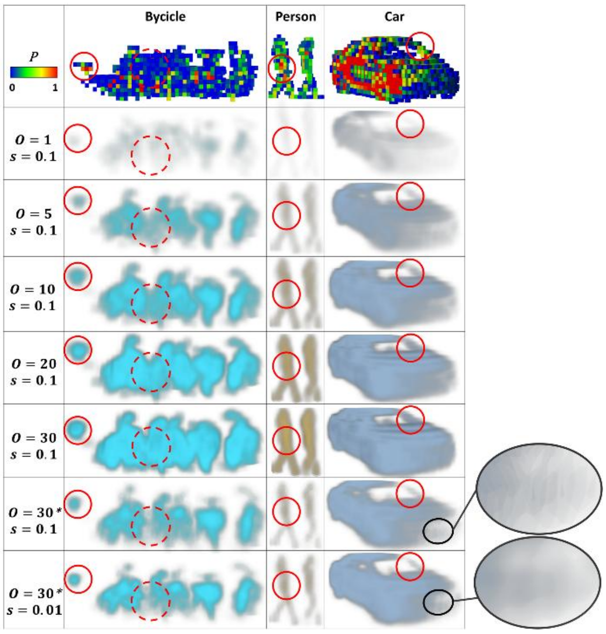

Figure 9. "Colorful Smoke" visualization for different O and s parameters.

For the second experiment, we aimed to emphasize voxels of high probability while reducing the visual confusion caused by overlapping low-probability voxels. A probability threshold with 0.1 intervals was applied, and the scaling factor O was additionally scaled in 0.1 increments, starting from O = 30. Consequently, probability values in the range 0.9–1.0 were sampled with O = 30, 0.8–0.9 with O = 30 × 0.9, and progressively decreasing in the same manner, down to O = 30 × 0.1 for probabilities below 0.1. Results are shown in Figure 9, marked with a star (∗). The effect is particularly evident in the "bicycle" and "person" clusters: high-probability regions are clearly distinguished by color intensity, while low-probability areas appear more transparent, even when overlapping. During this experiment, a banding pattern was observed due to high density variation. To reduce this, the sampling step s was decreased from 0.1 (equal to the voxel size) to 0.01, increasing the number of samples per voxel tenfold. This adjustment captures sharper probability gradients and produces smoother opacity transitions, as illustrated in Figure 9 ("car" cluster), inside zoomed black ovals highlighting the difference for O = 30∗ before (s = 0.1) and after (s = 0.01).

The resulting UDT is shown in Figure 10 and Figure 11. Static objects, including roads/terrain, buildings, trees, fences, street-lights, traffic signs, and bushes, are represented as voxels. Temporally occupied space by dynamic objects is visualized using the "Colorful Smoke" metaphor, with parameters O = 10 and s = 0.1 chosen here as illustrative examples. For visual assessment, Figure 10 presents an ego-centric view, while Figure 11 provides a bird's-eye perspective. In Figure 10 temporally occupied space can be clearly distinguished from permanently occupied areas through their distinct representations. The "Colorful Smoke" metaphor conveys temporal occupancy via a smoke-like, transient appearance, encodes object class through color, and represents occupancy probability using density and transparency. Additionally, Figure 10 highlights detailed building facades resulting from the fusion of CityGML and point cloud data, whereas Figure 11 provides an overview of the scene completed with CityGML and DTM data.

6. Conclusion and Outlook

This research introduces the "Colorful Smoke" visual metaphor for temporal space occupancy visualization employing a voxel based urban digital twin. The approach involves generating a voxel-based urban digital twin using CityGML, DTM, and point cloud data fusion. The dynamics of urban space are visualized through "Colorful Smoke", which communicates temporal occupancy by representing the density of smoke for occupied space, transparency for the likelihood of a space being occupied, and color for the type of dynamic object occupying the space. This visualization enables users to clearly identify areas of change in the urban environment and intuitively assess the reliability of the information without visual overload. Nevertheless, it is important to acknowledge the potential limitations of using the proposed visualization alongside realistic representations of natural phenomena, such as fire smoke simulation for emergency scenarios. The coexistence of metaphoric and realistic elements may cause users to confuse symbolic occupancy cues with actual fire smoke. To mitigate this, textures and color schemes specific to element type could be applied to differentiate the contexts.

The current occupancy-probability estimation relies on point counts, which may be biased by occlusions. Future work will consider including sensor fusion data and uncertainty modeling to reduce potential bias and strengthen the robustness of the approach. In terms of visualization, further experiments will aim to enhance the "Colorful Smoke" metaphor by introducing additional colors and performing a deeper analysis and adjustment of rendering parameters (O, s) to improve representation and better differentiate high-probability areas from low-probability regions in overlapping voxels. Additionally, a user study will be conducted to evaluate the intuitiveness and perceptual effectiveness of the proposed visualization in conveying temporal space occupancy. Finally, the urban digital twin will be simulated with frequent updates from dynamic data to explore the performance of "Colorful Smoke" in continuously changing urban scenes.

References

Bass, C. J., Anderson, E. F., 2013. Smoke and shadows: Rendering and light interaction of smoke in real-time rendered virtual environments. 2013 *5th International Conference on Games and Virtual Worlds for Serious Applications*, 1–4.

Besl, P., McKay, N., 1992. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239–256. doi.org/10.1109/34.121791.

Boccardo, P., Riccia, L. L., Yadav, Y., 2024. Urban Echoes: Exploring the Dynamic Realities of Cities through Digital Twins. *Land*, 13(5), 635. doi.org/10.3390/land13050635.

Bourke, P., 1999. Nearest neighbour weighted interpolation.

Cabello, R., contributors, T., 2010–2025. Three.js [Computer software]. https://threejs.org/. Accessed: 2025-08-15.

Campello, R., Moulavi, D., Sander, J., 2013. Density-based clustering based on hierarchical density estimates. J. Pei, V. Tseng, L. Cao, H. Motoda, G. Xu (eds), *Advances in Knowledge Discovery and Data Mining, PAKDD 2013*, Lecture Notes in Computer Science, 7819, Springer, Berlin, Heidelberg, 160–172.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W19-2025-127-2026

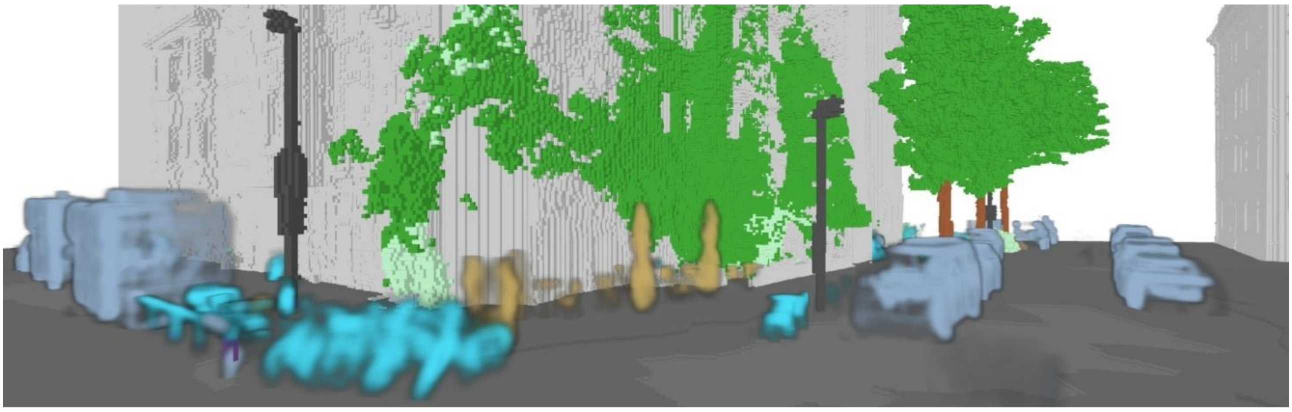

Figure 10. Voxel-based UDT (ego-centric view) with "Colorful Smoke" visualization (O = 10, s = 0.1).

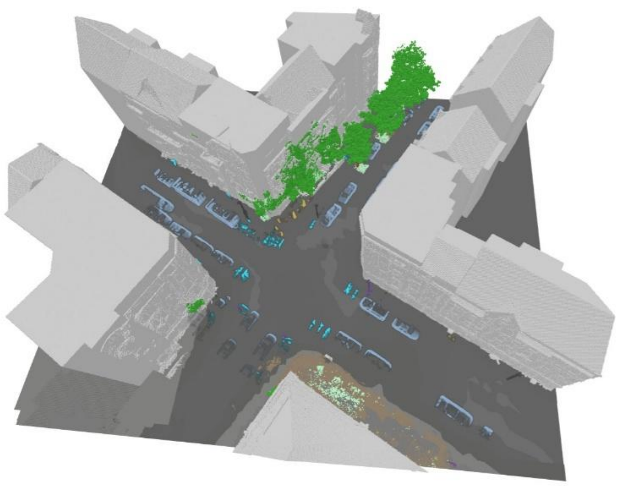

Figure 11. Bird's-eye view of the complete voxel-based UDT scene generated from CityGML, DTM, and point cloud data fusion.

Chehrehbargh, F. J., 2022. Fishermans Bend Digital Twin. https://eng.unimelb.edu.au/csdila/projects/digital-twin/fishermans-bend. Accessed: 2025-08-15.

Chew, L., 1987. Constrained Delaunay triangulations. 215–222. doi.org/10.1145/41958.41981.

City of Helsinki, 2025. Helsinki 3D Mesh – Urban Digital Twin. https://kartta.hel.fi/3d/mesh/. Accessed: 2025-09-16.

City of Rotterdam, 2025. 3D Rotterdam. https://www.3drotterdam.nl. Accessed: 2025-08-15.

Coumans, F., 2019. The Rotterdam 3D City Model. *GIM International*. https://www.gim-international.com/content/article/digital-city-rotterdam. Accessed: 2025-08-15.

Dembski, F., Wössner, U., Letzgus, M., Ruddat, M., Yamu, C., 2020. Urban Digital Twins for Smart Cities and Citizens: The Case Study of Herrenberg, Germany. *Sustainability*, 12(6), 2307. doi.org/10.3390/su12062307.

Ferré-Bigorra, J., Casals, M., Gangolells, M., 2022. The adoption of urban digital twins. *Cities*, 131, 103905. doi.org/10.1016/j.cities.2022.103905.

Fischler, M., Bolles, R., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381–395. doi.org/10.1145/358669.358692.

Forceville, C., 1996. Pictorial Metaphor in Advertising. doi.org/10.4324/9780203064252.

Fuller, A., Fan, Z., Day, C., Barlow, C., 2020. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access*, 8, 108952–108971. doi.org/10.1109/ACCESS.2020.2998358.

Hart, J., 1996. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*.

Herman, L., Russnák, J., Řezník, T., 2017. Flood modelling and visualizations of floods through 3d open data. J. Hřebíček, R. Denzer, G. Schimak, T. Pitner (eds), Environmental Software Systems. *Computer Science for Environmental Protection. ISESS 2017*, IFIP Advances in Information and Communication Technology, 507, Springer, Cham, 123–134.

Hugues, 2019. KPConv: Kernel Point Convolution. https://github.com/Hugueshttps/KPConv. Accessed: 2025-08-15.

Jeddoub, I., Nys, G.-A., Hajji, R., Billen, R., 2023. Digital Twins for cities: Analyzing the gap between concepts and current implementations with a specific focus on data integration. *International Journal of Applied Earth Observation and Geoinformation.* doi.org/10.1016/j.jag.2023.103440.

Khronos Group, 2024. OpenGL Shading Language (GLSL) specification. https://www.khronos.org/opengl/. Accessed: 2025-08-15.

Kumar, K., Ledoux, H., Stoter, J., 2018. Dynamic 3D Visualization of Floods: Case of the Netherlands. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-4/W10, 83–87. doi.org/10.5194/isprs-archives-XLII-4-W10-83-2018.

Levoy, M., 1988. Display of Surfaces from Volume Data. *IEEE Computer Graphics and Applications.*

Lin, H., Guo, R., Ma, D., Kuai, X., Yuan, Z., Du, Z., He, B., 2024. Digital-twin-based multi-scale simulation supports urban emergency management: a case study of urban epidemic transmission. *International Journal of Digital Earth*, 17(1). doi.org/10.1080/17538947.2024.2421950.

Martella, A., Ramadan, A. I. H. A., Martella, C., Patano, M., Longo, A., 2023. State of the art of urban digital twin platforms. *Extended Reality: International Conference, XR Salento 2023, Lecce, Italy, September 6-9, 2023, Proceedings, Part I*, Springer-Verlag, Berlin, Heidelberg, 299–317.

Mortazavi, F. S., Shkedova, O., Feuerhake, U., Brenner, C., Sester, M., 2023. Voxel-Based Point Cloud Localization for Smart Spaces Management. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLVIII-1/W1-2023, 325–332. doi.org/10.5194/isprs-archives-XLVIII-1-W1-2023-325-2023.

Niedersachsen Landesvermessungsamt, 2019. ALKIS - Amtliches Liegenschaftskataster-Informationssystem. https://www.lgln.niedersachsen.de/startseite/. Accessed: 2025-08-15.

Niedersachsen, O., 2019. OpenGeoData Niedersachsen. https://ni-lgln-opengeodata.hub.arcgis.com/pages/opengeodata. Accessed: 2025-08-15.

PostGIS, 2025. PostGIS. http://postgis.net/. Accessed: 2025- 08-15.

PostgreSQL, 2025. PostgreSQL. http://www.postgresql.org/. Accessed: 2025-08-15.

Rauer-Zechmeister, S., Cornel, D., Sadransky, B., Horváth, Z., Konev, A., Buttinger-Kreuzhuber, A., Heidrich, R., Blöschl, G., Gröller, E., Waser, J., 2024. HORA 3D: Personalized Flood Risk Visualization as an Interactive Web Service. *Computer Graphics Forum*. doi.org/10.1111/cgf.15110.

Ren, J., Nakata, S., 2024. A Fast and Memory-Efficient Direct Rendering Method for Polynomial-Based Implicit Surfaces. *Computer Modeling in Engineering and Sciences*, 141(2), 1033–1046. doi.org/10.32604/cmes.2024.054238.

Schrotter, G., Hürzeler, C., 2020. The Digital Twin of the City of Zurich for Urban Planning. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 88. doi.org/10.1007/s41064-020-00092-2.

Shkedova, O., Sester, M., 2024. Visualization of Space Occupancy Uncertainty in a 3D Voxel-based Urban Model. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci*., X-4-2024, 311–317. doi.org/10.5194/isprs-annals-X-4-2024-311-2024.

Somanath, S., Naserentin, V., Eleftheriou, O., Sjölie, D., Wästberg, B. S., Logg, A., 2024. Towards Urban Digital Twins: A Workflow for Procedural Visualization Using Geospatial Data. *Remote Sensing,* 16(11), 1939. doi.org/10.3390/rs16111939.

Steinicke, F., Mensmann, J., Hinrichs, K., Buhr, J., Rothaus, K., Krüger, A., 2008. Augmenting 3d city models with visualization of real-time meteorological phenomena. *Proceedings of the Third International Conference on Computer Graphics Theory and Applications*, 359–366.

Teles, B., Mariano, P., Santana, P., 2020. Game-Like 3D Visualisation of Air Quality Data. Multimodal Technologies and Interaction, 4(3), 54. doi.org/10.3390/mti4030054.

Vanella, M., McGrattan, K., McDermott, R., Forney, G., Mell, W., Gissi, E., Fiorucci, P., 2021. A Multi-Fidelity Framework for Wildland Fire Behavior Simulations over Complex Terrain. *Atmosphere*, 12(2), 273. doi.org/10.3390/atmos12020273.

Vetter, M., 2023. The usage of "fog" as a visual variable in 3D-cartographic virtual reality environments. *Abstr. Int. Cartogr. Assoc.*, 6, 263. doi.org/10.5194/ica-abs-6-263-2023.

Westover, L. A., 1991. SPLATTING: A parallel, feed-forward volume rendering algorithm. https://www.cs.utah.edu/lw/papers/splatting.pdf. Accessed: 2025-08-15.

Wijayawardena, A., Abeysekera, R., Maduranga, M., 2023. A Systematic Review of 3D Metaphoric Information Visualization. *International Journal of Modern Education and Computer Science (IJMECS)*, 15(1), 73-93. doi.org/10.5815/ijmecs.2023.01.06.

Würfel, H., Trapp, M., Limberger, D., Döllner, J., 2015. Natural phenomena as metaphors for visualization of trend data in interactive software maps. *CGVC*, 69–76.

Wysocki, O., Schwab, B., Hoegner, L., Kolbe, T., Stilla, U., 2021. Plastic surgery for 3d city models: a pipeline for automatic geometry refinement and semantic enrichment. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci*., V-4-2021, 17–24.

Yao, Z., Nagel, C., Kunde, F., et al., 2018. 3DCityDB – a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. *Open Geospatial Data, Software and Standards*, 3, 5. doi.org/10.1186/s40965-018-0046-7.

Zamri, M. N., Sunar, M. S., 2014. Real-Time Atmospheric Cloud Visualization Framework for Urban Simulation System. 2, 30-34.

Zhang, Y., Zou, L., Liu, J., 2011. Simulation of snow effects in visual simulation of virtual campus based on osg. *2011 International Conference on Multimedia Technology (ICMT 2011).*

Zhou, K., Ren, Z., Lin, S., Bao, H., Guo, B., Shum, H.-Y., 2008. Real-time smoke rendering using compensated ray marching. *ACM Trans. Graph*., 27(3), 1–12. doi.org/10.1145/1360612.1360635.