Game world generation based on geospatial data

Selthon Noronha de Moura[1], Jucimar Maia da Silva Junior[1], Cristina Souza de Araujo[1]

[1]Escola Superior de Tecnologia – Universidade do Estado do Amazonas (UEA)
Caixa Postal 1200 – 69065-020 – Manaus – Amazonas – Brasil

{sndm.snf22,csdaraujo}@uea.edu.br, jjunior@gmail.com

Abstract. Introduction: *Integrating geospatial data into digital games enables maps and environments that reflect real-world spatial structures. Instead of manually modeling terrain and cities, developers increasingly rely on datasets that describe geographic features such as roads, buildings, and terrain elevation. This trend is driven by improvements in game engines and geospatial data processing tools.* ***Objective:*** *To describe the main techniques and technologies used for generating maps in digital games based on real geospatial data.* ***Methodology:*** *This work was conducted through a review of academic literature and official documentation of widely adopted tools in the field. Aspects such as supported data formats, integration with game engines, licensing, and applicability were analyzed.* ***Results:*** *The survey identifies commonly adopted solutions and workflows in game development, along with legal and technical challenges related to data usage. It also highlights research opportunities in efficiently and creatively using spatial data in games.*
Keywords *Geospatial data, Urban digital twins, 3D city modeling, Procedural generation, Digital games.*

1. Introduction

In the field of digital games, the creation of realistic virtual worlds has become an increasing focus, driving a demand for the faithful reproduction of real-world locations. One emerging approach to achieve this involves the use of geospatial datasets in the generation of virtual environments. Instead of manually modeling complex terrains, road networks, or city layouts, developers may rely on geographic data to define the spatial logic and organization of game environments. This approach is supported by open data availability and modern tools that simplify geospatial integration into virtual environments.

Geospatial data has been applied in location-based games, flight simulators, and racing games, where the authenticity of the environment directly impacts the player's experience. In these and other similar genres, the accuracy of geographic and architectural elements increases the sense of realism provided to users. Major titles invest in this approach to provide immersive experiences, transporting players to virtual versions of real-world locations with varying levels of integration.

Other franchises, such as city simulation games, apply advanced procedural generation techniques to create virtual cities that follow orders and structures inspired by real-world patterns. This method allows developers to build highly detailed environments in an automated way, where the simulated cities visually resemble the real ones. Despite growing adoption, challenges remain, including data integration, performance optimization, and licensing restrictions.

Given this scenario, the present study conducts a survey of the main technologies used in the generation of maps for digital games using geospatial data. The objective is to identify and describe the main tools, workflows, and approaches currently adopted in the field, analyzing their capabilities, integration potential with game engines, and legal considerations. The goal is to provide a consolidated overview of existing solutions, synthesize academic and industrial contributions, and identify research gaps that can guide future investigations.

2. Background

Surveys are a fundamental mechanism for organizing knowledge, particularly in multidisciplinary and evolving fields such as game development. They help researchers and practitioners understand existing techniques within a common framework, identify underexplored approaches, and map opportunities for innovation and improvement [Mulrow 1994].

In the context of this study, a proper understanding of geospatial data and its integration into digital games is essential. Therefore, this section presents key definitions and concepts that underpin the rest of the survey. These include the nature and structure of geospatial data, its uses in virtual environment creation, and the role of Geographic Information Systems (GIS) in managing and transforming spatial information.

These foundations provide the necessary context for analyzing the technologies and tools explored in subsequent sections, especially regarding their integration with game engines and their potential for generating realistic and immersive game worlds.

2.1. Geospatial data

Geospatial data refers to information that is associated with specific locations on the Earth's surface. This type of data includes coordinates (latitude, longitude, and altitude) and may contain descriptive attributes such as land use, infrastructure, vegetation, and building typologies. It is commonly collected through satellite imagery, aerial photography, LiDAR, and GPS technologies and is usually stored in vector or raster formats with spatial referencing systems [Longley et al. 2015].

In the context of digital game development, geospatial data serves as a foundation for building virtual environments that reflect real-world locations. Using publicly available datasets or commercial platforms, developers can create virtual worlds directly from real data. This practice not only enhances the accuracy of the recreation and enables new types of gameplay, such as location-based and simulation-based games.

In digital games, the adoption of digital twin concepts enables the creation of urban spaces with a high level of realism and contextual accuracy. Game engines can simulate large cities that behave in ways similar to their real-world counterparts, enabling deeper immersion in open-world and simulation-based titles.

2.2. GIS

Geographic Information Systems (GIS) are frameworks for capturing, managing, analyzing, and visualizing spatial or geographic data. These systems link geographic coordinates with attribute information, allowing spatial analysis and decision making across various domains such as environmental monitoring, urban planning, public health, and transportation logistics [Burrough e McDonnell 1998].

GIS software can process both vector data (points, lines, polygons) and raster data (gridded formats such as satellite imagery or elevation models), facilitating operations such as map overlay, buffering, spatial interpolation, and network analysis. GIS allows users to cross-reference different layers of information, such as land use, population density, and hydrology.

In addition to processing existing datasets, modern GIS platforms also allow users to acquire geospatial data directly through plugins and external connections. Examples include Web Map Services (WMS), Web Feature Services (WFS), spatial database connections, and institutional or open-data APIs. Through this capability, GIS can work as a central hub for integrating and curating data from multiple sources.

Platforms such as QGIS, ArcGIS, and GRASS GIS can serve as preparatory environments where raw geospatial data from multiple sources can be cleaned, transformed, and exported for further use in external applications. For developers interested in incorporating real-world topographies or administrative boundaries into their games, GIS provides the necessary tools to filter and refine this information with precision and context-awareness.

2.3. Location-Based Games

Location-Based Games (LBGs) are digital games in which gameplay is influenced by the player's physical location, often detected via GPS or other geolocation technologies. These games merge virtual content with real-world spatial contexts, encouraging players to interact with their surrounding environment as part of the game experience [da Silva et al. 2024, Coelho et al. 2019].

LBGs are typically characterized by features such as geolocation tracking, map-based interfaces, and context-aware mechanics. Examples of popular LBGs include Pokémon GO, Ingress, and Jurassic World Alive, which use geospatial data to generate in-game events, spawn locations, and objectives tied to real-world coordinates.

The development of LBGs relies heavily on the integration of geospatial data sources, spatial querying, and visualization techniques. As such, they represent a compelling intersection between game design, cartographic systems, and location-based services. Understanding LBGs is useful for contextualizing the application of spatial technologies in modern game development.

3. Methodology

This work is structured as a technical survey aimed at identifying, categorizing, and describing technologies used in the generation of maps in digital games based on geospatial data. The survey was conducted through a review of academic literature and technical documentation to map the current state of the field.

The process was divided into three main stages:

1. **Scope definition**: The guiding research question was established as: *Which tools and techniques are being used to create realistic virtual environments in digital games based on geospatial data?*

2. **Source collection and selection**: Searches were conducted across the following academic databases: IEEE Xplore, ACM Digital Library, and CAPES Portal. Subsequently, the official documentation and technical portals of each selected technology were reviewed to gather detailed information.
   To ensure relevance and quality, sources were selected based on the following criteria:
   - The source described a technology explicitly used or usable in game development or immersive environments.
   - The tool or technique made use of geospatial data (e.g., satellite imagery, elevation models, vector maps).
   - The publication or documentation was published or last updated within the last two years, except for foundational works cited in more recent developments.
   - The source was accessible in English and provided sufficient technical detail to assess functionality and application.
3. **Data extraction and organization**: The collected information was organized by technology, highlighting aspects such as (i) types of geospatial data supported; (ii) integration with game engines; (iii) licensing requirements; and (iv) use cases in game development or immersive applications.

The result of this process is the systematization of a representative set of technological solutions applicable to urban modeling in digital games, with emphasis on their technical viability, integration potential, and practical applicability.

4. Survey

In this section, we describe the main technologies currently available for integrating geospatial data into digital game development. Rather than providing a comparative evaluation, the goal is to highlight representative tools and platforms that enable the creation of spatially informed game maps.

Each subsection focuses on a specific technology, outlining its functionality, integration options, data compatibility, and common use cases. Together, they reflect the growing ecosystem of solutions at the intersection of geospatial data and interactive media.

4.1. Cesium Ion

Cesium Ion is a 3D geospatial visualization platform that enables the real-time display of 3D regions with realistic representations of the real world, using geospatial data. For gaming, the platform offers free and open-source packages for integration with 3D game engines such as Cesium for Unreal and Cesium for Unity, which bring Cesium's 3D geospatial capabilities into these respective game development environments. This integration allows developers to create immersive applications, games, and simulations using real-world data to provide highly detailed virtual environments.

Cesium's applicability for rapid game environment creation has been demonstrated in practice. In the post-mortem of the game *Galactic Gourmet*, developed during a game jam, the authors describe how Cesium was used to generate the virtual

Figure 1. City of Manaus visualized through Cesium Ion

environment automatically from real-world geospatial data, significantly reducing manual level design effort [Ahmed e Noor 2020].

The Cesium for Unreal and Cesium for Unity packages work by automatically inserting a 3D tileset from Cesium Ion servers into the scene, based on a latitude, longitude, and altitude georeference inputted by the developer as the origin point. This allows dynamic loading of 3D tiles covering the entire region visible to the in-game camera.

4.2. OpenStreetMap

OpenStreetMap (OSM) is a collaborative open-source mapping project that provides free geographic data for various applications, including games and urban simulations. Through the official OSM website, users contribute to the database by adding streets, points of interest, and buildings using the editing mode.

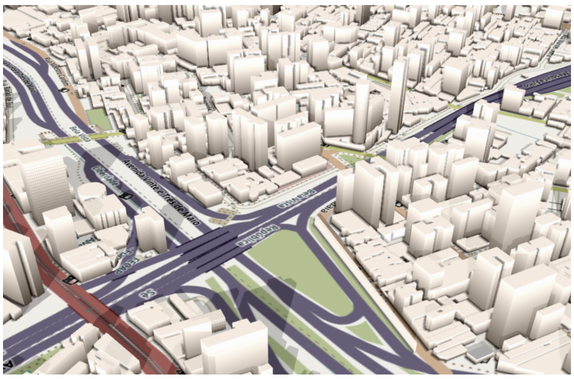

Figure 2. Buildings from the city of São Paulo reconstructed using OSM data

Map elements in OSM can have multiple semantic tags, allowing for correct interpretation when converting data into virtual environments. These data can be queried and extracted through the Overpass API, an interface for customized queries to the OSM database. Using these tags, developers can identify elements such as houses, schools, hospitals, parks, roads, bridges, and bike lanes, facilitating the automated creation of urban maps in interactive applications. Tools such as osmnx, osm2world, and plugins for game engines make it even easier to transform this data into 3D physical representations.

OSM is a favored data source for location-based games and mobile apps due to its accessibility and public data model. It has been used in games like Pokémon GO and

Jurassic World Alive. The database is licensed under the Open Data Commons Open Database License (ODbL) v1.0, which allows both commercial and non-commercial use with proper attribution.

The potential of OSM for procedural generation has been explored in the context of city-building games. [Pinos et al. 2019] proposed a pipeline to automatically transform OSM data into playable environments in Cities: Skylines, accurately recreating the cities of Oulamoc and Svit with minimal manual intervention, using semantic tags to generate roads, zoning, and buildings in line with the original urban layout.

4.3. Bing Maps

Bing Maps is Microsoft's mapping service, offering geolocation data, satellite imagery, and APIs for integrating interactive maps into apps and games. Its map database includes data from OpenStreetMap, supplemented with exclusive Bing Maps data, including satellite imagery worldwide. The Bing Maps API consists of five APIs accessed through a single key: V8 Web Control, Windows 10 Universal Windows Platform, Windows Presentation Foundation, REST Services, and Spatial Data Services.

Microsoft uses Bing Maps data directly in its flight simulation game, achieving realistic reproductions of real-world cities that players can freely fly over with high detail. Due to its association with OpenStreetMap data, players wishing to increase the detail of cities in the game can contribute directly by adding geographic information through the online platform.

Bing Maps satellite imagery has been integrated into research projects involving urban digital twins. [Rantanen et al. 2023] used Bing Maps satellite images in conjunction with Cesium Ion to replicate a Finnish city inside Unreal Engine.

The use of Bing Maps APIs and data is governed by Microsoft's Terms of Service and requires proper attribution. A basic API key can be created for free, but it comes with a transaction limit.

4.4. Procedural Generation

Procedural generation refers to algorithmic techniques for automatically creating content and is commonly used in city simulation games to build environments reminiscent of real cities. It is also used to create digital twins of real cities, especially when spatial data is incomplete, such as the positioning of trees, buildings, lampposts, and other smaller details typically not included in geospatial datasets [Smelik et al. 2014, Parish e Müller 2001].

Procedural city generation can be achieved through various methods such as rule-based systems, noise algorithms (e.g., Perlin and Simplex), formal grammars (e.g., L-systems), and graph-based systems to simulate streets, blocks, and urban zones [Wonka et al. 2003, Parish e Müller 2001]. These approaches allow for the creation of complex and varied structures based on defined parameters or small input datasets.

It is also possible to combine incomplete geospatial data with procedural algorithms to fill in missing elements and generate details that increase the realism of the environment. For example, a street map obtained via OSM can be enhanced with buildings, vegetation, and urban furniture generated procedurally, following architectural or cultural patterns of the represented region. This technique is especially valuable in large-scale games or projects that require real-time generation, such as flight simulators or open-world games, where manual environment modeling would be unfeasible.

4.5. ArcGIS CityEngine

ArcGIS CityEngine is an advanced 3D modeling tool developed by Esri, specifically designed for creating detailed cityscapes and urban environments. Its main differentiator is its ability to procedurally generate buildings, streets, and urban landscapes based on geospatial data and customizable rules. This approach enables the efficient creation of large urban areas that are visually coherent, making it ideal for urban planning projects, simulations, architectural visualizations, and applications in games.

ArcGIS CityEngine integrates with the ArcGIS ecosystem and supports importing geographic data such as shapefiles, GeoJSON files, or even OpenStreetMap data. These datasets can be used to automatically generate realistic 3D environments based on real spatial attributes.

Although its use in commercial games is still limited, ArcGIS CityEngine has been explored in projects focused on recreating detailed real-world locations from GIS datasets, such as the Vltava River valley [Janovský 2024], that can be exported to game engines such as Unity and Unreal Engine, enabling its integration into digital game development.

The software is proprietary and requires a paid license, especially for commercial use. Esri offers various licensing options, including individual, institutional, and educational licenses, depending on the use case and project scale. Specific terms, limitations, and rights are outlined in Esri's official Terms of Use.

4.6. Mapbox

Mapbox is a versatile mapping platform that provides developers with tools to integrate real-world geospatial data into applications, including games. Through its Maps SDK for Unity, Mapbox enables the creation of interactive, location-based experiences by offering access to global map data, customizable styles, and real-time updates.

The SDK supports features such as dynamic map rendering, terrain elevation, and point of interest (POI) data, which can be utilized to design immersive game environments. Developers can leverage Mapbox's vector tiles and satellite imagery to generate detailed 3D maps that respond to user interactions and real-world changes.

Mapbox's flexibility allows for the development of various game genres, including augmented reality (AR) games, racing simulators, and strategy games that require accurate geographic representations. Its integration with Unity facilitates the incorporation of GPS data, enabling the creation of games that adapt to the player's physical location.

In [Metikaridis e Xinogalos 2020], in order to compare Mapbox with similar tools for developing location-based games, the authors develop an LBG called *Pocket Droids Go* using the Mapbox Maps SDK for Unity. The game features the procedural generation of a real-world map and incorporates geolocation mechanics and interaction scenes entirely rendered within Unity.

The platform offers a tiered pricing model, including a free tier suitable for prototyping and small-scale projects. Comprehensive documentation and an active developer community support the implementation of Mapbox in game development workflows.

5. Comparative Overview

Table 1 provides a comparative summary of the technologies presented in this survey. It outlines key aspects such as the types of geospatial data supported, integration capabilities with game engines, licensing restrictions, and common applications in game development. This overview highlights the diversity of tools available and assists in identifying suitable technologies depending on the project scope and technical requirements.

Table 1. Comparative overview of geospatial technologies for game world generation.

Technology	Data Types	Integration with Game Engines	Licensing	Use in Games
Cesium Ion	3D Tiles, terrain, satellite imagery	Unity, Unreal (official plugins)	Commercial license	Simulations, AR/VR, immersive city views.
OpenStreetMap	Vector (Points, Lines and Areas)	External APIs and tools (osmnx, osm2world, Unity plugins)	Open Database License (ODbL 1.0)	Location-based games, generation based on tagged data.
Bing Maps	Raster, vector, satellite imagery	Unity via REST APIs or third-party SDKs	Microsoft license, limited by API key	Location-based games, Real-world terrain rendering, and simulation scenarios
Mapbox	Vector tiles, raster imagery, terrain elevation, POIs	Unity (via official Maps SDK), web-based tools	Free tier available, commercial license for extended use	AR games, location-based experiences, custom 3D map visualizations
CityEngine	Shapefiles, GeoJSON, Vector	Exports to Unity/Unreal via FBX, OBJ	Commercial license	3D city modeling, experimental game scenarios.

6. Scoring Matrix

To complement the descriptive comparison presented in Table 1, Table 2 introduces a scoring matrix that provides an analytical assessment of the technologies based on technical criteria. The matrix considers seven key dimensions relevant to game development workflows, described in the legend below.

Each technology is evaluated on a scale from 1 (low/limited) to 5 (high/excellent) for each criterion. The goal is not to produce an absolute ranking, but rather to highlight trade-offs and strengths, offering a clearer picture of which tools may be more appropriate for specific use cases in game development.

Legend:

- **R (Rendering)** – Real-time 3D rendering capability and visual quality.
- **M (Memory)** – Efficiency in memory usage and resource management.
- **I (Integration)** – Ease of integration with game engines such as Unity or Unreal.

Table 2. Comparative Scoring Matrix for Game-Oriented Geospatial Technologies

Technology	R	M	I	L	P	C	D
Cesium Ion	4	2	3	2	3	3	5
OpenStreetMap	2	5	3	5	5	5	2
Bing Maps	3	3	2	3	1	2	5
ArcGIS CityEngine	5	3	2	2	5	3	4
Mapbox	4	3	5	3	2	2	4
Procedural Gen.	3	5	2	5	1	3	1

- **L (License)** – Flexibility and permissiveness of licensing for commercial and non-commercial use.
- **P (Procedural)** – Support for or compatibility with procedural content generation.
- **C (Community)** – Availability of developer support, documentation, and community resources.
- **D (Global Data)** – Access to embedded or pre-hosted global geographic datasets.

6.1. Justification of Scores by Technology

6.1.1. Cesium Ion

Cesium Ion received high scores in rendering and global data availability due to its efficient streaming of 3D Tiles and high-quality photogrammetric datasets. Its integration with Unity and Unreal is functional but requires adaptation for custom gameplay logic. Licensing depends highly on the datasets being used, and the tool's community support is growing. While it supports procedural enhancement through dynamic LOD loading, full procedural generation features are limited.

6.1.2. OpenStreetMap

OpenStreetMap stands out in terms of licensing freedom, community strength, and support for procedural generation. Although it does not offer built-in rendering engines, its data can be adapted for real-time rendering through third-party tools. It performs well in memory usage due to its lightweight vector data and excels in extensibility through semantic tagging.

6.1.3. Bing Maps

Bing Maps offers high-quality satellite imagery and reliable coverage of global data. However, its licensing is more restrictive, and its integration with game engines is less flexible. It does not natively support procedural generation, and its community engagement is less active compared to open-source platforms.

6.1.4. ArcGIS CityEngine

ArcGIS CityEngine is highly specialized in procedural modeling, offering advanced CGA rule systems for generating urban environments. It provides excellent rendering quality and supports data import from various GIS sources. However, its commercial license is restrictive for indie developers, and integration into game engines requires export workflows. Documentation is solid, but community support is more academic than game-oriented.

6.1.5. Mapbox

Mapbox provides seamless integration with Unity through its SDK, offering dynamic maps, elevation data, and POI layers. It scores high in integration and rendering, but has moderate licensing flexibility and limited procedural features. The platform benefits from strong documentation and developer resources, especially for location-based applications.

6.1.6. Procedural Generation

As a technique rather than a standalone tool, procedural generation scores highly in memory efficiency and creative flexibility. It complements incomplete datasets and allows dynamic generation of complex content. The topic is well supported by academic literature, libraries, and game development communities. However, it requires manual implementation or integration with other systems. It also lacks native geospatial data or direct rendering capabilities.

7. Conclusion

The integration of geospatial data into game development workflows has opened new possibilities for creating dynamic, data-driven virtual environments. Real-world spatial data allows developers to automate the generation of coherent, context-aware worlds. This reduces manual effort and enhances gameplay realism.

This survey mapped the main technologies and tools currently available for this purpose, analyzing their data formats, engine integration capabilities, licensing models, and typical use cases. Technologies such as Cesium, OpenStreetMap, Bing Maps, ArcGIS CityEngine, and procedural generation pipelines offer complementary approaches, each with unique strengths depending on the scale, genre, and design goals of the game being developed.

However, the adoption of these technologies still faces practical barriers, including the complexity of data preprocessing, performance concerns with large-scale environments, and legal issues surrounding data reuse and attribution. These challenges highlight the need for more unified workflows, standardization across tools, and better documentation for integration with modern game engines. In particular, bridging the gap between GIS platforms and game development environments remains a technical and conceptual challenge that limits broader adoption.

Future work could explore the development of solutions that bridge GIS platforms and game engines, enabling real-time geospatial data streaming, procedural enhancement of partial datasets, or AI-assisted interpretation of urban features. Efforts in procedural city generation, semantic tagging of spatial assets, and machine learning-based generation of plausible missing data could further strengthen the integration between geospatial datasets and game design.

Additionally, broader adoption of open data policies and advances in interoperability may expand the creative possibilities for developers working at the intersection of geography and interactive media. Such developments could benefit not only games for entertainment, but also serious applications in education, urban planning, disaster simulation, and location-based storytelling. As spatially grounded virtual experiences become more pervasive, fostering dialogue between the fields of geography and game development will be crucial to building tools and standards that support this emerging design space.

8. Acknowledgements

The author thanks Professor Jucimar Maia da Silva Junior for his dedicated guidance and Professor Cristina Souza de Araujo for her co-supervision and technical support. The author also thanks the LUDUS Lab for providing structure for the execution of the research. This work was developed within the scope of the Arkade Academy project, as part of the RD&I ARKADEVERSE initiative, carried out by the Amazonas State University (UEA) in partnership with Tectoy S/A, using resources from Federal Law No. 8.387/1991. Its dissemination and publicity are in accordance with the provisions of Article 39 of Decree No. 10.521/2020. The author also thanks all colleagues and institutions who contributed directly or indirectly to the completion of this project.

References

Ahmed, S. e Noor, F. A. (2020). Rapid game world creation leveraging geospatial data in game jams: The ‘galactic gourmet’ post-mortem. In *Proceedings of the 15th International Conference on the Foundations of Digital Games*.

Beam, C., Zhang, J., Kakavitsas, N., Hague, C., Wolek, A., e Willis, A. (2024). Cesium tiles for high-realism simulation and comparing slam results in corresponding virtual and real-world environments. *arXiv*.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., e Çöltekin, A. (2015). Applications of 3d city models: State of the art review. *ISPRS International Journal of Geo-Information*.

Brol, A. e Antoniuk, I. (2023). Procedural generation of virtual cities. Technical report, Institute of Information Technology, Warsaw University of Life Sciences.

Burrough, P. A. e McDonnell, R. A. (1998). *Principles of Geographical Information Systems*. Oxford University Press.

Coelho, A., Cardoso, P., Camilo, M., e Sousa, A. (2019). Designing of a mobile app for the development of pervasive games. *2019 International Conference on Graphics and Interaction (ICGI)*.

da Silva, B. C., Freitas, L. M., Maia, J. G. R., e Viana, W. (2024). Identifying patterns and affordances in location-based games: The practices of niantic. In *XXIII Simpósio Brasileiro de Jogos e Entretenimento Digital*.

Gaisbauer, W., Raffe, W. L., Garcia, J. A., e Hlavacs, H. (2019). Procedural generation of video game cities for specific video game genres. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*.

Gevaert, C. M., Buunk, T., e van den Homberg, M. J. C. (2024). Auditing geospatial datasets for biases: Using global building datasets for disaster risk management. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Gröger, G. e Plümer, L. (2012). Citygml – interoperable semantic 3d city models. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Janovský, M. (2024). Pre-dam vltava river valley—a case study of 3d visualization of large-scale gis datasets in unreal engine. *ISPRS International Journal of Geo-Information*, 13(10):344.

Li, Z., Wegner, J. D., e Lucchi, A. (2019). Topological map extraction from overhead images. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Longley, P. A., Goodchild, M. F., Maguire, D. J., e Rhind, D. W. (2015). *Geographic Information Science and Systems*. John Wiley & Sons.

Metikaridis, D. e Xinogalos, S. (2020). A comparative analysis of tools for developing location based games. *Entertainment Computing*, 37:100403.

Mulrow, C. D. (1994). Systematic reviews: rationale for systematic reviews. *BMJ*, 309(6954):597–599.

Over, M., Schilling, A., Neubauer, S., e Zipf, A. (2010). Generating web-based 3d city models from openstreetmap: The current situation in germany. *Computers, Environment and Urban Systems*.

Parish, Y. I. e Müller, P. (2001). Procedural modeling of cities. *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, page 301–308.

Pepe, M., Costantino, D., Alfio, V. S., e Grimaldi, A. (2024). An effective method for 3d modelling of urban areas from aerial images and lidar data. *ISPRS Archives*.

Pinos, J., Vozenilek, V., e Pavlis, O. (2019). Automatic geodata processing methods for real-world city visualizations in cities: Skylines. *ISPRS International Journal of Geo-Information*.

Rantanen, T., Julin, A., Virtanen, J., Hyyppä, H., e Vaaja, M. T. (2023). Open geospatial data integration in game engine for urban digital twin applications. *ISPRS International Journal of Geo-Information,*.

Smelik, R. M., Tutenel, T., Bidarra, R., e Benes, B. (2014). A survey on procedural modelling for virtual worlds. *Computer Graphics Forum*, 33(6):31–50.

Somanath, S., Naserentin, V., Eleftheriou, O., Sjölie, D., Wästberg, B. S., e Logg, A. (2023). Towards urban digital twins: A workflow for procedural visualization using geospatial data. Technical report, Chalmers University of Technology.

Suveg, I. e Vosselman, G. (2003). Reconstruction of 3d building models from aerial images and maps. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Varinlioglu, G., Vaez Afshar, S., Eshaghi, S., Balaban, , e Nagakura, T. (2022). Gis-based educational game through low-cost virtual tour experience- khan game. In *27th International Conference of the Association for Computer-Aided Architectural Design Research in Asia: Post Carbon*, pages 69–78.

Wonka, P., Wimmer, M., Sillion, F., e Ribarsky, W. (2003). Instant architecture. *ACM Transactions on Graphics*, 22(3):669–677.

Ziegler, M. (2012). *QGIS: A Free and Open Source Geographic Information System*, pages 107–122. Springer.