

EstateXR: An Augmented Reality-Based Real Estate Platform

Abhinav Singh | Under the guidance of Dr. Prateek Raj Gautam

Department of Computer Science and Engineering, School of Computer Science University of
Petroleum and Energy Studies, Dehradun-248007, Uttarakhand

Abstract - Property discovery and architectural planning still rely heavily on static 2D floor plans, CAD blueprints, and pre-rendered walkthroughs, which leave a persistent gap between how a space is depicted and how it is actually experienced. Existing virtual-tour platforms extend this model but remain descriptive: they can show what already exists rather than project a proposed structure onto vacant land. This paper proposes EstateXR, a browser-based Augmented Reality (AR) framework intended to close that gap using WebXR/WebAR technologies (AR.js and WebGL) delivered through a React.js and Node.js front end with a Three.js rendering engine. The design centres on three components: a Spatial-Topographic Alignment layer for anchoring 3D models to real-world GIS coordinates and terrain, a Modular "Creator" layer for interactive, drag-and-drop structural design, and a GIS-integrated land data layer. We outline the architecture, the algorithmic building blocks (sensor-fusion tracking, point-cloud terrain mapping, level-of-detail asset streaming, and spherical-harmonics lighting estimation), and the expected trade-offs of each. As this is a design and architecture paper rather than a deployed-system evaluation, we present the framework as a reference model and identify the prototype-and-user-study work required to validate it empirically.

1. INTRODUCTION

The global real estate and construction industry is at an inflection point in how physical spaces are marketed and planned. Since the early digital transition of the 2000s, property discovery has relied on static 2D imagery, CAD blueprints, and pre-rendered video walkthroughs. As spatial computing matures, these methods are increasingly insufficient for a market that expects interactive, remote-access visualization.

This gap is rooted in what we refer to as the cognitive-spatial gap: the difficulty of mentally translating a flat 2D floor plan into an accurate sense of a 3D physical environment. Commercial platforms such as Zillow and Matterport popularised 360-degree virtual tours, but these remain descriptive rather than prescriptive - they show existing structures rather than project a proposed, non-existent building onto a vacant plot in real time. Professional-grade alternatives built on Building Information Modelling (BIM) or Geographic Information Systems (GIS) address parts of this problem but require specialised hardware and engineering expertise, which puts them out of reach for individual homebuyers, retail investors, and small developers.

A further limitation of many mobile AR applications is technical rather than conceptual: models can drift from their anchor point as a user moves (anchor drift), or appear at the wrong scale relative to the real plot (scaling discrepancy). Addressing both the accessibility gap and these tracking issues within a single, browser-based system is the central design problem this paper addresses.

We propose EstateXR, an AR platform intended to operate through standard mobile browsers rather than a dedicated app or headset, using AR.js and WebGL for rendering and tracking, a React.js/Node.js front end, and Three.js for 3D rendering. The intent is to let a user move from browsing a listing to viewing an AR visualization with no installation step.

2. RELATED WORK

Academic and industry interest in AR/VR for real estate has grown over the past decade, though most published work targets marketing and buyer engagement rather than construction-grade spatial accuracy. Gallardo et al. developed iMovBusca, a mobile AR application for real estate classified ads that superimposes listing information such as price and address onto a live camera feed; a usability study in that work found users' AR- and map-mode preferences varied by task, underlining that AR value depends on what information is being conveyed, not only on the visualization itself.

Flavián, Ibáñez-Sánchez and Orús reviewed how virtual, augmented, and mixed reality technologies shape customer experience more broadly, providing a framework for why immersive property visualization can influence buyer confidence. Jung et al. explored a "try before you buy" AR concept that let users reposition virtual furniture or change room finishes in real time, reporting that this kind of interactivity meaningfully affected purchasing intent. Lang and Sittler's early work on AR for real estate similarly focused on overlaying property information rather than generative or modular structural design.

Most of this literature treats AR as an overlay on an existing, already-built property. EstateXR's point of departure is the prescriptive case: projecting a building that does not yet exist onto a vacant, unmapped plot, which raises the anchor-stability and terrain-alignment problems that overlay-only systems do not need to solve. This positions the present work as an architectural proposal for that gap rather than a replication of existing marketing-oriented AR tools.

3. PROBLEM DEFINITION

Modern property discovery is typically limited to static 2D blueprints and fragmented GIS data, which are prone to scaling errors and misaligned buyer expectations. Even 360-degree tours and pre-rendered walkthroughs do not address the harder case of projecting a non-existent, modular structure onto vacant terrain. Doing so within a mobile browser - while maintaining 1:1 scale accuracy, resisting anchor drift, and accounting for uneven terrain - is the specific technical problem this paper defines and proposes an architecture for.

4. OBJECTIVES

- Construct a WebAR reference architecture that integrates real-time spatial computing with a browser-based interface, without requiring native app installation.
- Establish a taxonomy of spatial-visualization errors - anchor drift, scaling mismatch, and topographic clipping - and relate each to a specific component of the AR pipeline.
- Examine the environmental and hardware constraints (lighting sensitivity, GIS data inconsistency, device variability) relevant to on-site mobile AR.
- Propose a multi-tier architecture linking GIS land data with a Three.js rendering pipeline for low-latency 3D superimposition on uneven terrain.
- Define the platform's core features - a 3D asset repository, an interactive plot selector, and a bespoke creator suite - as independently deployable services.

5. PROPOSED ARCHITECTURE

5.1 Spatial Tracking and Topographic Alignment

The Spatial Tracking layer combines gyroscope, accelerometer, and magnetometer input (sensor fusion) to establish a stable local coordinate frame for the AR session. The Topographic layer is responsible for ground-plane estimation: since a real plot is rarely a flat Euclidean surface, this layer converts point-cloud data captured from the device camera into a terrain mesh, so that a building's foundation follows the plot's actual slope rather than floating above it or clipping into it.

5.2 Modular Creator Suite

The Creator module is intended as an interactive sandbox in which a user positions walls, windows, and roof elements directly within the AR viewport. A decoupled design is proposed, in which the UI layer captures parametric input (dimensions, materials) while the Three.js scene graph updates independently, so that structural changes are reflected with minimal delay.

5.3 Hardware-Agnostic Delivery

To keep the platform usable on a typical smartphone rather than requiring a headset, Level-of-Detail (LOD) asset streaming is proposed so that high-polygon models are swapped for simplified versions as needed, and the design targets compatibility across the WebXR/WebGL implementations found in current mobile browsers.

5.4 GIS-Integrated Land Data

A geospatial mapping service is proposed to host land coordinate and ownership data and to supply the plot boundary that the AR viewer anchors to. The reliability of this layer - specifically the accuracy of survey data and GPS positioning - is a direct input to the accuracy of the 1:1 scale overlay, and is treated in this paper as an external dependency rather than something the AR layer alone can guarantee.

6. ALGORITHMIC BUILDING BLOCKS

- Spatial tracking and anchor stability: sensor fusion (e.g., an Extended Kalman Filter combining gyroscope and accelerometer data) to reduce jitter from rapid camera movement.
- Topographic alignment and occlusion: point-cloud-to-mesh conversion to align a building's virtual foundation with the physical slope of the land.

- Proportional scaling: coordinate transformation between camera focal distance and the virtual anchor to preserve 1:1 scale.
- Asset management: frustum culling and Level of Detail (LOD) to limit rendering cost to visible geometry.
- Lighting estimation: spherical-harmonics-based estimation from camera frames to match shadow direction to real sunlight.

Ray-casting for precise interaction within the Creator module, and CNN-based surface/boundary recognition for unmapped terrain, are noted as plausible extensions rather than components already validated in this work.

7. ADVANTAGES AND LIMITATIONS

7.1 Advantages

- Combines GIS-based land selection, 1:1 scale AR overlay, and interactive design in one framework rather than as separate tools.
- WebAR/WebGL delivery avoids the need for a headset or native app install, lowering the hardware barrier to entry.
- Multi-sensor tracking is intended to reduce anchor drift relative to single-sensor AR implementations.
- Terrain- and lighting-aware rendering is intended to reduce visible scaling and clipping errors before they reach the end user.
- A modular Creator/Visualizer split allows each service to be updated or scaled independently.

7.2 Limitations

- The architecture integrates several distinct technology stacks (WebGL rendering, AR tracking, GIS data protocols), which raises implementation and maintenance complexity.
- Real-time light estimation and high-poly asset streaming carry ongoing compute and bandwidth cost.
- WebXR and AR.js standards are still evolving, so cross-browser behaviour (Chrome vs Safari) may require ongoing maintenance.
- Visualization accuracy depends on external GIS data quality and GPS precision, both of which are outside the platform's direct control.
- Integrating with legacy property-listing databases and differing 3D asset formats (glTF, OBJ) adds interoperability overhead.

8. SUMMARY OF COMPONENTS

The table below summarises the proposed components, their associated algorithms, the results expected from each (pending empirical validation), and known limitations.

Method Used	Algorithm Used	Expected Result	Future Development	Limitations
Spatial Tracking Layer	Sensor Fusion (Extended Kalman Filter)	Improved anchor stability and reduced drift versus single-sensor tracking	Integration of LiDAR for millimetre-level precision	High dependence on device IMU quality
Topographic Alignment	Point-cloud mesh mapping	Closer 1:1 scale alignment on uneven terrain	Real-time slope and elevation analysis	Surface detection degrades on low-texture areas
Modular Creator Suite	Decoupled scene graph	Real-time architectural customization in the AR viewport	AI-driven interior decor suggestions	Higher memory usage on mid-range devices
Asset Optimization	Level of Detail (LOD)	Lower latency and steadier frame rate	Automated high-to-low poly compression	Visual popping during asset swaps
Environmental Lighting	Spherical Harmonics	More realistic shadow and light matching	Dynamic Global Illumination (DGI)	Battery drain from continuous camera processing
Geospatial Mapping	GIS coordinate synchronisation	More precise plot boundary identification	Marketplace API for real-time pricing	Inconsistencies in local land databases
User Access Layer	WebXR API	App-less, browser-based entry	Cross-platform AR cloud persistence	Browser version compatibility issues

9. CONCLUSION AND FUTURE WORK

This paper has outlined EstateXR, a proposed browser-based AR architecture for real estate visualization that emphasises three components largely absent from existing commercial tools: topographic alignment for uneven terrain, a modular structural Creator suite, and hardware-agnostic delivery through WebXR. The framework is presented as a design contribution rather than a validated system: the immediate next step is to implement a working prototype of the tracking and alignment layers and to run a controlled user study - comparing task accuracy and user confidence against a conventional 2D-blueprint baseline - to test the claims made here about reduced spatial misinterpretation. Future work should also address LiDAR-assisted tracking for precision, AI-assisted interior design suggestions within the Creator module, and integration with live GIS/marketplace pricing data.

REFERENCES

- [1] Gallardo, C., Rodríguez, S. P., Chango, I. E., Quevedo, W. X., Santana, J., Acosta, A. G., & Andaluz, V. H. (2018). Augmented Reality as a New Marketing Strategy. In *Augmented Reality, Virtual Reality, and Computer Graphics: 5th International Conference, AVR 2018, Otranto, Italy (Part I)*, pp. 351-362. Springer.
- [2] Gallardo, C., et al. Using and Evaluating Augmented Reality for Mobile Data Visualization in Real Estate Classified Ads. *International Journal of Computers and Applications*, 36(1), 7-14.
- [3] Flavián, C., Ibáñez-Sánchez, S., & Orús, C. (2019). The impact of virtual, augmented and mixed reality technologies on the customer experience. *Journal of Business Research*, 100, 547-560.
- [4] Jung, T., et al. (2015). Augmented reality and "try before you buy" in real estate: an exploratory study of user experience with virtual furniture placement.
- [5] Lang, V., & Sittler, P. (2012). Augmented reality for real estate. Research paper.
- [6] Li, Z., & Liu, H. Technological advancements of virtual realities for research in real estate marketing. In *Augmented and Virtual Reality Property Visualization and Enhanced Client Communication*. Springer Nature.
- [7] Mozilla / Immersive Web Working Group. WebXR Device API Specification. <https://www.w3.org/TR/webxr/>
- [8] Three.js Documentation. <https://threejs.org/docs/>
- [9] AR.js Documentation. <https://ar-js-org.github.io/AR.js-Docs/>