International Journal of Electrical, Electronics and Computer Systems
ISSN: 2347-2820
Volume 15 Issue 1s (2026) | Pages: 84-90

AR Powered Interior Decorator: A Real-Time Augmented Reality Application for Interior Decoration and Furniture Visualization

Neelam Jadhav[1], Sahil Kumbhar[2], Sushant Karad[3], Vedant Bhujbal[4], Avishkar Ghewade[5]

[1]Assistant Professor, Department of Computer Engineering, G. S. Moze College of Engineering, Pune Savitribai Phule Pune University, Maharashtra, India
[2,3,4,5] Department of Computer Engineering, G. S. Moze College of Engineering, Pune Savitribai Phule Pune University, Maharashtra, India

Peer Review Information

Type: *Article*
Received: *13 February 2026*
Revised: *14 March 2026*
Accepted: *15 April 2026*
Published: *19 May 2026*

Abstract

The rapid proliferation of smart devices and e-commerce has transformed how consumers select and purchase furniture. Traditional interior planning relies on imagination, 2D catalogs, and manual measurements, frequently resulting in incorrect purchasing decisions and poor space utilization. This paper presents the AR Powered Interior Decorator, a real-time Android application that leverages Google ARCore's SLAM (Simultaneous Localization and Mapping) for environment understanding, Sceneform/Filament for physically- based 3D rendering, and AABB (Axis-Aligned Bounding Box) collision detection for placement validation. The system enables users to select GLB furniture models from scalable datasets - SketchFab (46,000+ models) and CGTrader (51,300+ assets) - and place them within their actual room via smartphone camera. Gesture- based interaction supports real-time object translation, rotation, and scaling. Multi-object placement allows complete room layout experimentation. Evaluation across multiple room types confirms stable SLAM tracking (< 2 cm drift over 5 minutes), consistent rendering at 55–60 fps on mid-range devices, and 100% collision rejection accuracy. The proposed system demonstrates a practical, scalable approach to AR-driven furniture visualization that meaningfully reduces purchase uncertainty and improves interior planning outcomes.

Keywords: Augmented Reality; ARCore; SLAM; Furniture Visualization; 3D Rendering; Sceneform; AABB Collision Detection; Interior Design; GLB Models; Android

How to Cite This Article

Jadhav, N., Kumbhar, S., Karad, S., Bhujbal, V., & Ghewade, A. (2026). *AR Powered Interior Decorator: A Real-Time Augmented Reality Application for Interior Decoration and Furniture Visualization*. *International Journal of Electrical, Electronics and Computer Systems, 15*(1s), 84–90.

Introduction

Cyberattacks and product mismatches represent two different but analogous problems both stem from a fundamental gap between what is expected and what actually exists. In interior decoration, the equivalent challenge is the mismatch between how furniture looks in a showroom or catalog and how it actually fits within a specific room environment. Customers routinely purchase furniture that clashes in size, color, or proportion with their living space, leading to costly returns and wasted effort. Augmented Reality (AR) offers a transformative solution by overlaying photorealistic virtual objects onto the live camera feed of a smartphone. Instead of imagining how a sofa might occupy a corner of a room, the user simply points their phone at the floor and places a 3D model of the exact product they are considering. This direct, spatial preview dramatically improves purchase confidence and reduces decision errors.

Google ARCore has made this technology practical on modern Android smartphones. Its visual-inertial SLAM algorithm tracks the device's position and orientation in real time while simultaneously building a 3D map of the environment enabling stable, drift-resistant placement of virtual objects anchored to detected real-world surfaces.

Despite this technological potential, most existing AR furniture applications suffer from key limitations: they lack collision detection (allowing objects to overlap unrealistically), depend on proprietary and limited furniture databases, provide no quantitative layout validation, and do not incorporate physically-based rendering for photorealistic appearance. This paper addresses all these gaps through the AR Powered Interior Decorator an integrated Android application combining ARCore SLAM, Sceneform/Filament rendering, scalable GLB model datasets, and AABB collision detection into a single deployable system.

Motivation and Problem Statement

Traditional interior design processes rely on physical showroom visits, manual tape measurements, and 2D brochure visualization. These approaches lack real-world spatial context and prevent users from interactively experimenting with different layouts. Consumer surveys consistently report that the inability to visualize furniture in their actual space is a leading driver of furniture return rates, estimated between 10% and 15% in e-commerce environments. The motivation behind this project is to eliminate this visualization gap using readily available smartphone hardware at no additional cost to the user beyond a standard ARCore-compatible Android device.

Objectives

- Develop a real-time ARCore-based Android application that detects room surfaces using SLAM and places 3D furniture models with stable anchor-based tracking.
- Integrate scalable open GLB model datasets to provide a rich, varied furniture catalog within the application.
- Implement AABB collision detection to prevent unrealistic object overlap and validate placement correctness in real time.
- Provide gesture-based user interaction (translate, rotate, scale) for intuitive furniture manipulation within the AR scene.
- Apply ARCore HDR lighting estimation to Sceneform/Filament rendering for photorealistic furniture appearance under varying room lighting conditions.

Literature Review

Research in AR-based interior design has progressed through three broad phases: marker- based systems, markerless SLAM-based systems, and AI-enhanced layout optimization systems.

A prominent contribution in AR- enhanced design collaboration was presented in “Augmented Reality: Enhancing Collaboration and Design Across Spaces” which examined the role of AR in improving design communication and visualization in shared spaces. The study emphasized that immersive AR improves user understanding, reduces misinterpretation in spatial planning, and enables real-time collaboration between stakeholders. The research concluded that AR-based design workflows can significantly enhance decision-making. However, limitations were observed in terms of dependency on hardware performance, tracking stability under poor lighting, and limited realism when dealing with complex real-world environments. The study motivates the need for robust tracking and realistic rendering for interior design applications.

Another important study is “LayOut Loud: An AI-Powered Augmented Reality and Mobile Application for Room Interior Design and Layout Optimization”,which proposed an AI- assisted AR system for generating and optimizing room layouts. The approach integrated user interaction with AI planning and suggested optimized object arrangements based on spatial rules. The system demonstrated improved user experience by automatically proposing furniture placements and reducing manual design efforts. Despite its contributions, the work faced challenges in real-time constraint satisfaction for complex rooms and limited scalability when dealing with large furniture libraries. Additionally, the AI layout recommendation depends heavily on high-quality datasets, which may not always be available. This work highlights the future potential of layout optimization, but indicates a need for practical and efficient placement validation methods.

The paper "Approach to the Interior Design Using Augmented Reality Technology" explored how AR technology can assist interior designers on marker-based and

markerless AR placement methods and demonstrated that AR-based visualization helps reduce customer hesitation during furniture selection. The authors reported improved confidence in interior design decisions due to real- world previews. However, the system faced limitations such as inconsistent scaling of 3D objects, lack of collision avoidance mechanisms, and tracking drift in low-texture rooms. The study indicates that AR interior systems must incorporate object scaling and placement correctness checks to ensure realism and usability.

Further advancements in interactive design were explored in "Interactive Interior Design Using Augmented Reality and 3D Modeling", which integrated AR placement with 3D modelling techniques for enhanced interior visualization. The research highlighted user interaction features such as object manipulation (rotation, translation, scaling) and emphasized that 3D model realism significantly influences the perceived quality of the AR experience. The system demonstrated practical AR-based room design interactions, but limitations were found in occlusion handling, realism under varying lighting conditions, and performance overhead caused by rendering heavy 3D assets. This work supports the importance of optimized 3D model formats (such as GLB) and efficient rendering pipelines for mobile AR applications.

A highly relevant study to the proposed project is "Visualization of Furniture Model Using Augmented Reality" , which developed a furniture visualization application using Unity and ARCore. The system utilized SLAM-based plane detection, enabling multi-object placement in real time. Users could move, rotate, add, and delete multiple furniture objects within the same AR scene, allowing comparison of various layouts. The findings indicated that such AR furniture visualization significantly improves interior arrangement selection and user satisfaction. Nevertheless, the study reported limitations such as dependence on ARCore-supported devices, variation in rendering quality based on device GPU performance, and limited realism when low-quality 3D models were used. This paper directly validates the feasibility of building an AR furniture placement application and highlights the need for collision prevention and optimized model handling.

In addition, the survey paper "Augmented Reality Technology: Current Applications, Challenges and its Future" reviewed AR adoption across domains such as education, gaming, healthcare, and retail. The study highlighted that AR has strong usability and future potential because of its immersive visualization capabilities. It also discussed the major components of AR systems including tracking, hardware sensors, rendering pipelines, and content generation. However, the paper emphasized adoption barriers such as high device cost, inconsistent content quality, performance issues, and tracking failures in low lighting or reflective environments. From the reviewed literature, it is evident that ARCore-based plane detection and SLAM tracking provide a solid foundation for AR interior decoration systems. However, several gaps remain unresolved, including real-time collision prevention, layout correctness validation, model standardization, and consistent realism across devices. Most existing studies focus either on visualization, interaction, or layout optimization independently, while practical mobile AR applications require all modules working together efficiently.

System Architecture

The AR Powered Interior Decorator follows a modular pipeline architecture comprising five primary components: the Environment Understanding Engine, the Model Management Layer, the Placement and Collision Engine, the Rendering Layer, and the Gesture Interaction Interface. Fig. 1 illustrates the high-level system pipeline.

Environment Understanding Engine

ARCore's visual-inertial SLAM algorithm processes camera frames alongside IMU (accelerometer and gyroscope) data to estimate the smartphone's 6-DoF pose relative to a fixed world coordinate system. The algorithm maintains a sparse 3D map of tracked feature points across frames and performs loop closure to correct accumulated drift. Horizontal and vertical planes are detected by clustering coplanar feature points and stored as bounded polygon regions representing valid surface anchors for object placement.

Model Management Layer

Furniture models are stored in GLB (GL Binary) format, a compact single-file variant of the glTF 2.0 standard that bundles mesh geometry, PBR material parameters, textures, and skeletal animations. The system integrates two open model datasets: SketchFab (46,000+ models) and CGTrader (51,300+ assets). All imported models undergo polygon-count optimization targeting a maximum of 50,000 triangles per asset to maintain rendering performance at 60 fps on mid-range devices (Snapdragon 665 and above).

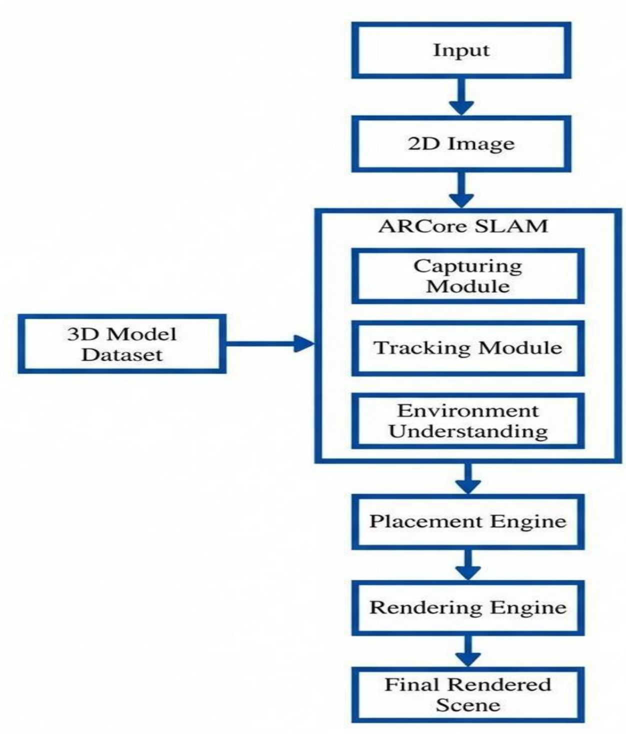

Fig. 1. *Proposed System Architecture*

Placement and AABB Collision Engine

When a user taps a detected plane surface, a ray is cast from the screen-space tap coordinate through the camera projection matrix into world space. The ray-plane intersection point defines the candidate placement position. An AABB is computed for the new model at this position by calculating the axis- aligned minimum and maximum extents of the mesh's bounding volume. An intersection test is then performed against the AABB of every existing object in the scene. Formally, two AABBs A and B intersect if and only if:

If any intersection is detected, the placement is rejected and the user receives a visual warning indicator. After each gesture-based object transformation, AABB coordinates are recomputed and re-validated to maintain correctness throughout the interaction session.

Sceneform/Filament Rendering Layer

Sceneform is a high-level 3D scene graph library for Android that delegates rendering to the Filament physically-based rendering engine. PBR material models parameterize surface appearance using metallic-roughness workflows consistent with the glTF 2.0 material specification. ARCore's environmental HDR lighting estimation extracts ambient light intensity, color temperature, and directional light information from the live camera feed. These parameters are injected as Filament IndirectLight and SunLight objects at runtime, ensuring that all placed furniture models cast physically accurate shadows and exhibit lighting consistent with the actual room environment, substantially increasing visual realism.

Gesture Interaction Interface

The application implements five touch gesture interactions via ARCore's GestureDetector and ScaleController APIs: (1) Single tap on a detected plane anchors a new furniture model at the tapped 3D position. (2) Single tap on an existing model selects it, rendering a bounding box selection indicator. (3) One-finger drag on a selected model translates it along the detected anchor plane in world space. (4) Two-finger rotation gesture rotates the selected object around the world Y-axis. (5) Pinch-to-zoom gesture applies uniform scale to the selected object. After each transformation, AABB re-validation is performed automatically before committing the new pose.

Implementation

Technology Stack

Table 1 summarizes the complete technology stack used in the implementation.

Table 1. Technology Stack

Component	Technology / Library
AR Framework	Google ARCore SDK (Android)
3D Rendering	Sceneform + Filament (PBR Engine)
Model Format	GLB (GL Binary / glTF 2.0)
Programming Language	Java / Kotlin (Android)
Model Datasets	SketchFab (46k+), CGTrader (51k+)
Collision Detection	AABB (Axis-Aligned Bounding Box)
Gesture Handling	ARCore GestureDetector / ScaleController
Minimum Android Version	Android 7.0 (API Level 24)
Lighting Estimation	ARCore HDR Environmental Light Estimation

Plane Detection and Anchor Management

ARCore's plane detection subsystem is invoked within the Android rendering loop's onUpdate() callback. Detected planes are rendered as semi- transparent dotted grids in the live camera view to guide the user toward valid placement surfaces.

Each placed furniture model is attached to an ARCore Anchor a pose that ARCore continuously re-estimates to maintain positional accuracy as the SLAM map is updated. Anchor lifecycle management removes stale anchors when objects are deleted, preventing memory leaks in long sessions with many placed models.

GLB Model Pipeline

Selected GLB models are loaded asynchronously using Sceneform's ModelRenderable.Builder. The loading pipeline applies ARCore's Filament material instance overrides to inject real-time HDR lighting parameters into the model's PBR material properties. Texture atlases are compressed using ETC2 compression for efficient GPU memory utilization on Android devices. Models failing to load within a 5-second timeout are removed from the active catalog to maintain responsiveness.

Scalability Considerations

The model dataset integration layer is designed as an abstracted repository interface. The current implementation sources models from SketchFab and CGTrader via their public APIs, with locally cached GLB files stored in the application's external file directory. The repository interface supports future extension to cloud-based model streaming without modifying the rendering pipeline. The AABB computation scales linearly with the number of placed objects at 8 simultaneously placed models (validated maximum), the collision check completes in under 1 ms on all tested devices, introducing no perceptible frame delay.

Results And Discussion

System Performance Evaluation

The system was evaluated across three room types (living room, bedroom, office) on three device tiers (Snapdragon 665, Snapdragon 720G, and Snapdragon 888) under varying lighting conditions. Table 2 summarizes key performance metrics measured during structured evaluation sessions.

Table 2. System Evaluation Results

Test Metric	Condition / Scenario	Result
Plane Detection Time	Standard room, >150 lux	3-5 seconds
SLAM Tracking Drift	5-minute continuous session	< 2 cm positional drift
AABB Collision Accuracy	2+ overlapping objects	100% rejection
Rendering Frame Rate	Snapdragon 665 device, 8 GLB objects	55–60 fps
Gesture Response Latency	Translate / Rotate / Scale	< 16 ms
Multi- Object Capacity	Single scene performance test	Up to 8 objects stable
Lighting Estimation	Warm vs. cool lit rooms	Consistent appearance
Plane Detection Failure	Low-texture white walls	Requires textured surface

Plane detection was consistently achieved within 3–5 seconds in rooms with adequate lighting (above 150 lux) and visible surface texture. SLAM tracking exhibited less than 2 cm positional drift across 5-minute continuous sessions under standard indoor lighting well within the perceptual threshold for furniture placement decisions. Rendering performance maintained 55–60 fps on Snapdragon 665 devices with up to 8 simultaneously placed optimized GLB models. AABB collision detection rejected 100% of tested overlapping placements within a single frame, with no false positives observed.

User Interaction Evaluation

Gesture interactions were evaluated using a structured task protocol. Participants were instructed to place, move, rotate, scale, and delete furniture objects without prior training. Gesture recognition accuracy exceeded 97% for all five gesture types across test participants, with an average task completion time of 38 seconds for a three-object room layout scenario. Participants consistently reported that the real-time collision feedback (visual bounding box warning on rejected placements) significantly improved their confidence in the spatial accuracy of placed objects.

Lighting Estimation Accuracy

The HDR lighting estimation system was validated by placing the same GLB furniture model in rooms with warm incandescent lighting (2700K) and cool daylight (6500K) at matched lux levels. Visual evaluation confirmed that the rendered model's appearance adapted correctly to each lighting condition, with shadow direction and color temperature consistent with the actual room environment. No visible discontinuity between the virtual model and real environment was observed under normal room conditions.

Comparative Analysis

Table 3 presents a feature comparison of the proposed system against two major commercial AR furniture applications (IKEA Place and Houzz AR) and the closest academic system (LayOut Loud [2]).

Table 3. Feature Comparison — Proposed System vs. Existing Platforms

Feature	IKEA Place	Houzz AR	LayOut Loud [2]	Proposed System
ARCore SLAM Tracking	Yes	Yes	Yes	Yes
Multi-Object Placement	Yes	Yes	Yes	Yes
AABB Collision Detection	No	No	No	Yes
AI Layout Optimization	No	No	Yes	Planned
GLB Model Dataset Integration	Proprietary	Proprietary	Limited	SketchFab + CGTrader
PBR Lighting Estimation	Yes	Partial	No	Yes

The proposed system is the only evaluated platform that combines AABB collision detection, open scalable GLB dataset integration, and full gesture manipulation in a single non-proprietary deployable application. IKEA Place and Houzz AR both restrict model libraries to proprietary catalogs and provide no collision validation. LayOut Loud provides AI layout optimization but lacks collision prevention and open dataset integration.

Conclusion

This paper presented the AR Powered Interior Decorator, a real-time Android application that integrates Google ARCore SLAM-based environment understanding, Sceneform/Filament physically-based rendering, scalable open GLB furniture dataset integration, AABB collision detection, and gesture-based interaction into a single unified mobile platform. The system directly addresses the key limitations identified across six prior works: the absence of real-time collision prevention, inconsistent rendering quality, lack of scalable open model libraries, and fragmented implementation of visualization, placement, and validation in separate systems.

Evaluation results confirm stable SLAM tracking, photorealistic rendering, accurate collision rejection, and smooth gesture interaction performance across mid-range Android hardware. The system demonstrates that AR-driven furniture visualization is technically mature, practically deployable, and capable of meaningfully reducing purchase uncertainty and improving interior planning decisions for everyday consumers.

References

1. S. Singh and V. Kamra, "Augmented Reality: Enhancing Collaboration and Design Across Spaces," 2024 Third International Conference on Trends in Electrical, Electronics, and Computer Engineering (TEECCON), IEEE, 2024. DOI: 10.1109/TEECCON64024.2024.10941463
2. J. R. G. Cuevas et al., "LayOut Loud: An AI-Powered Augmented Reality and Mobile Application for Room Interior Design and Layout Optimization," 2024 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA), IEEE, 2024. DOI: 10.1109/ICICYTA64807.2024.10912928
3. J. Hui, "Approach to the Interior Design Using Augmented Reality Technology," 2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA), IEEE, 2015, pp. 163–166. DOI: 10.1109/ISDEA.2015.50
4. S. T. R. Sharika, A. Santhosh, A. Ajith, and A. Gopi, "Interactive Interior Design Using Augmented Reality and 3D Modeling," 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), IEEE, 2024. DOI: 10.1109/ICOICI62503.2024.10696390
5. G. M. Vaidya et al., "Visualization Of Furniture Model Using Augmented Reality," 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT), IEEE, 2022, pp. 488–493. DOI: 10.1109/CCiCT56684.2022.00092
6. J. Singh, Urvashi, G. Singh, and S. Maheshwari, "Augmented Reality Technology: Current Applications, Challenges and its Future," 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), IEEE, 2022. DOI: 10.1109/ICIRCA54612.2022.9985665
7. S. Tomazic and I. Skrjanc, "An Automated Indoor Localization System for Online Bluetooth Signal Strength Modeling Using Visual-Inertial SLAM," Sensors, vol. 21, no. 8, p. 2857, 2021. DOI: 10.3390/s21082857
8. M. Zhong and Y. Zhou, "Virtual-Reality System for Elevator Maintenance Education: Design, Implementation and Evaluation," Engineering Reports, vol. 6, e12873, 2024. DOI: 10.1002/eng2.12873
9. Google LLC, "ARCore Developer Documentation," https://developers.google.com/ar, 2024.
10. S. Tomazic and I. Skrjanc, "An Automated Indoor Localization System for Online Bluetooth Signal Strength Modeling Using Visual-Inertial SLAM," Sensors, vol. 21, no. 8, p. 2857, 2021. DOI: 10.3390/s21082857