Semantic Point-of-Interest–Driven IFC-Based Robot Navigation in Indoor Environments

Muhammad Shoaib Khan[1], Zhihao REN[2], Ho Jin Lee[1], Seonghyeun Kim[1], Woo-yong Jung[2], Jung In Kim[1]

[1] School of Civil and Environmental Engineering, Kookmin University, Seoul, 02707, Republic of Korea
[2] Department of Architecture and Civil Engineering, City University of Hong Kong, Kowloon, Hong Kong SAR
[3] Dept. of Nuclear Power Plant Engineering, KEPCO International Nuclear Graduate School, Ulsan, 45014, Korea
muhammadshoaib5308@gmail.com, z.ren@my.cityu.edu.hk, leehojin3239@kookmin.ac.kr, tjd1103@kookmin.ac.kr, trustjung@gmail.com, jikim07@kookmin.ac.kr

Abstract –

Autonomous mobile robots deployed in building environments require comprehensive spatial semantics for navigation and operation. However, while current Industry Foundation Classes (IFC) standards provide both geometric and semantic information, they lack robot-operational attributes. This paper presents a Point-of-Interest (POI) framework that enriches IFC building models with robot-oriented behavioural intelligence through function-based mapping rules. The proposed semantic enrichment methodology utilizes IfcOpenShell to augment standard IFC entities with custom property sets that encode POI definitions, thereby enabling the integration of building geometry and robot semantics within Unity-based virtual reality simulation environment. The framework is validated through various scenarios, including doorway accessibility assessment, multi-robot coordination, and wall-following tasks, demonstrating its effectiveness and applicability. The approach establishes a foundation for proactive robot-compatible building design and supports the development of automated accessibility evaluation tools.

Keywords –

BIM-IFC; Point of interest (POI), robot-friendly design, semantic enrichment, BIM to robot, virtual reality

1 Introduction

The deployment of autonomous mobile robots in building environments has become increasingly prevalent [1]. For robots to navigate effectively, they require comprehensive spatial understanding. Robots must identify navigable pathways, detect and avoid obstacles, recognize transition points between spaces, locate service stations for charging and maintenance, and coordinate movements with other robots and human occupants [2]. These operational requirements necessitate rich semantic information about the built environment that encodes not merely what spaces and objects are, but how they function in relation to robotic navigation and task execution [3].

Navigation environment modeling for autonomous systems is conventionally categorized into three types based on spatial representation: metric-based models (grid or voxel discretization), topology-based models (graph structures with nodes and edges), and network models that combine geometry with topology [4–6]. Metric-based approaches tessellate space into cells with pathfinding algorithms such as A* and Dijkstra applied for shortest path analysis, but suffer from computational overhead and storage requirements [7]. Topology-based models provide compact graph structures but lack the geometric precision necessary for operational robot guidance [4]. Network models integrate geometric coordinates with topological connectivity through representations including Visibility Graphs, Voronoi Diagrams, and Medial Axis Transforms, yet these geometry-derived methods face scalability challenges in complex building environments and struggle with irregular floor layouts [8].

Building Information Modeling (BIM) has emerged as the dominant framework for digitally representing building assets throughout their entire lifecycle, from initial design conception through construction execution to long-term operational management [9,10]. The Industry Foundation Classes (IFC) standard, maintained by BuildingSMART International, provides the foundational data schema for BIM interoperability, defining standardized representations for building elements, spatial structures, material properties, and functional relationships[11]. The prior studies have used

IFC data to construct navigation models, such as voxel-based maps, which are generated using IFC geometry [6], and a graph-based network for building-level connectivity is developed using IFC semantics [12].

However, current IFC specifications, while comprehensive for human-centric building design and construction management, exhibit fundamental limitations when applied to robot navigation and operation contexts. IFC schemas define building elements: walls, doors, floors, and spaces, primarily through properties relevant to architectural design, structural engineering, and building code compliance. Attributes such as wall thickness, door fire ratings, floor load capacities, and room occupancy classifications serve architectural and engineering purposes effectively, yet these properties do not directly address robotic operational requirements. Critical robot-navigation parameters remain largely absent from standard IFC property definitions. Consequently, existing BIM-driven navigation studies cannot directly support robot deployment planning, as the authoritative building information models do not contain the behavioral and operational attributes necessary to inform robot path planning, obstacle avoidance, and service execution algorithms.

Addressing this gap requires a systematic approach to the semantic enrichment of building information models with robot-oriented spatial intelligence [13]. Recent research has introduced Point-of-Interest (POI) frameworks as a mechanism for redefining building elements from robotic behavioral perspectives. Unlike geographic POIs that mark locations of human interest, such as landmarks or commercial establishments, robot-oriented POIs represent spatial elements that directly influence autonomous navigation decisions and task execution. A door, for instance, functions not merely as a physical barrier with dimensional properties, but as a transition element that governs spatial accessibility under specific clearance and passage constraints. A wall serves simultaneously as an obstacle requiring avoidance and as a reference boundary for path guidance. By assigning such behavioral roles to building elements, POI frameworks transform static architectural representations into semantically rich spatial models that encode how robots should perceive, interpret, and interact with their operational environments.

This paper presents a comprehensive framework for semantic POI-based robot navigation that leverages IFC through systematic semantic enrichment. By embedding POI behavioral roles as native IFC metadata, the framework enables a semantics-first approach where building elements carry robot-operational intelligence within the authoritative BIM model, directly informing navigation algorithms without requiring separate POI annotation systems or geometry-derived topology extraction. Virtual Reality simulation in Unity validates that POI-enriched IFC models provide actionable navigation guidance through sensor-based accessibility testing and iterative design refinement before physical construction.

2 Method

The proposed framework integrates BIM-based building representation, semantic POI enrichment, VR simulation, and robot navigation through a systematic methodology comprising four components. Figure 1 illustrates the overall system architecture and data flow. Each component maintains interoperability with standard BIM workflows while introducing robot-specific semantic intelligence necessary for autonomous navigation.

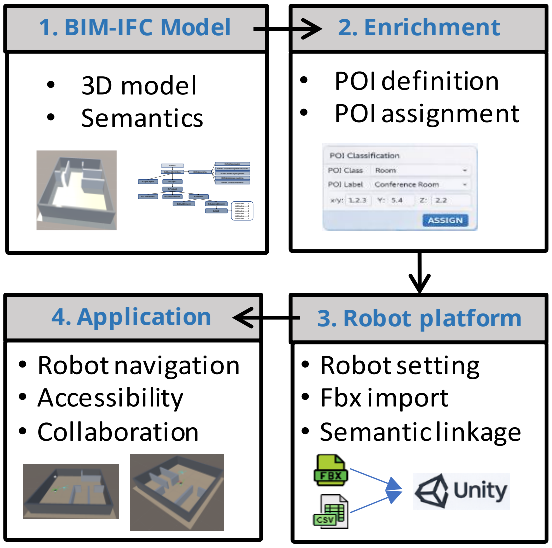


Figure 1. Proposed **F**ramework

2.1 BIM-IFC Building Model Development

Building information is represented as an IFC 4.3 file following standard BIM authoring practices, with building elements organized according to IFC hierarchical structure: IfcWall for boundary elements, IfcDoor for transition openings, IfcSpace for spatial zones (rooms, corridors), and IfcSlab for floor surfaces. IfcOpenShell, an open-source Python library for IFC file manipulation, is employed to parse the IFC model and extract both geometric and semantic information. Rather than developing custom IFC parsers, IfcOpenShell provides robust support for different IFC schema versions and enables straightforward extraction of element attributes through Python dot notation (e.g., IfcDoor.OverallWidth returns door width). The library's geometry processing capabilities convert implicit IFC geometric representations (swept geometry, boundary representations, compound solids) into explicit mesh data comprising vertices, edges, and faces with absolute coordinate placement, eliminating the complexity of handling relative IFC placement hierarchies.

For each IFC element, the extraction process captures: (1) Global Unique Identifiers (GUIDs) for element tracking, (2) geometric parameters including mesh vertices, faces, edges, and spatial coordinates, (3) semantic attributes and element classifications, and (4) spatial hierarchies. This structured dataset serves as the foundation for subsequent POI semantic enrichment, with element GUIDs providing persistent links between IFC entities and their robot-operational metadata.

2.2 Semantic Enrichment with POI Classifications

2.2.1 Formal Mapping Rules

The semantic enrichment process employs a function-based classification framework that systematically assigns POI roles to IFC entities based on their spatial and functional characteristics. This approach ensures framework generalizability across the complete IFC schema, with mapping rules applicable to related IFC entities possessing the relevant functional properties. Five core mapping rules govern POI assignment based on entity function: spatial boundary provision, space connection, spatial containment, fixed object presence, and runtime constraint detection. Table 1 presents the formal mapping rules with representative example entities illustrating rule application across structural, functional, and object-based POI categories.

Table 1: Formal POI mapping rules

Rules	Functional Condition	POI Assignment	Examples
R1	Provides spatial boundary	Structural POI (A-1, A-3)	IfcWall, IfcColumn, IfcCurtainWall, IfcRailing, IfcBeam
R2	Connects two spatial zones	Structural POI (A-2, B-2 if clearance <threshold)	IfcDoor, IfcWindow, IfcRamp, IfcStair, IfcOpening
R3	Defines a spatial container	Functional POI (A-3, C-1, B-1)	IfcSpace (any predefined type or name)
R4	Fixed object or equipment element	Object-based POI (A-1)	IfcFurniture, IfcEquipment, IfcDistribution Element
R5	Runtime constraint detected	Dynamic role addition (B-2, C-2)	Any element with active sensor feedback

2.2.2 POI Role Taxonomy

This research demonstrates framework feasibility through three core POI categories essential for autonomous navigation: A-Mobility, B-Safety, and C-Coordination. Table 2 defines the complete role taxonomy for these categories, specifying seven operational roles that govern robot behavioral responses to building elements during navigation.

Table 2 POI role taxonomy

Category	Code	Role Name	Robot Behaviour
A-Mobility	A-1	Avoidance	Maintain clearance
A-Mobility	A-2	Transition	Verify clearance, control passage speed
A-Mobility	A-3	Path Guidance	Use as a trajectory reference
B-Safety	B-1	Hazard Zone	Proceed with caution
B-Safety	B-2	Narrow/Congested	Reduce speed
C-Coordination	C-1	Multi-Robot Zone	Apply the coordination protocol
C-Coordination	C-2	Robot-Robot Interaction	Maintain separation, adjust speed

2.2.3 IFC Property Set Extensions

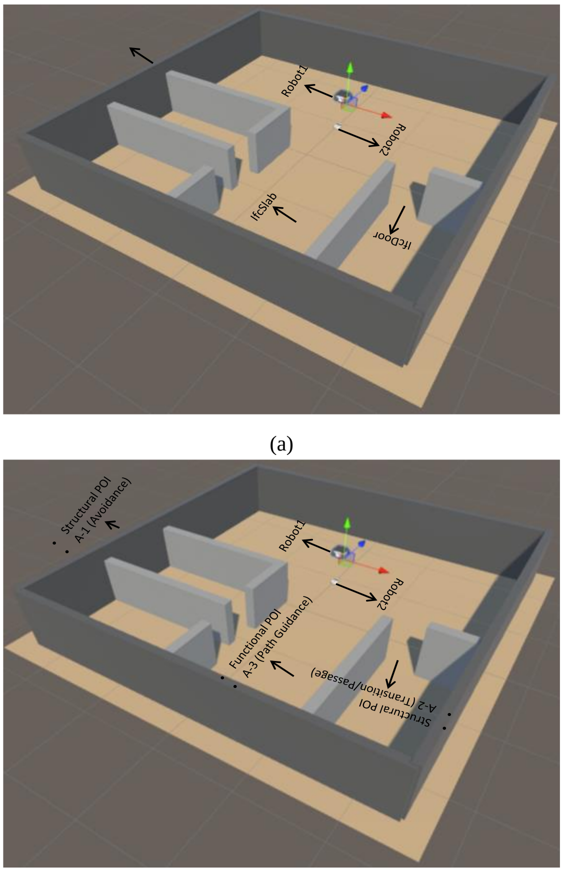
Three custom property sets are programmatically attached to IFC entities using IfcOpenShell property manipulation capabilities, extending standard IFC entities with robot-operational metadata while maintaining full backward compatibility with BIM authoring and viewing software. Pset_POI_Classification (attached to all relevant entities) defines POI_Category {Structural, Functional, Object-based}, POI_Type {Boundary Element, Transition Element, Space Type}, POI_PrimaryRoles [array of assigned role codes], and POI_ConditionalRoles [runtime-assignable roles based on sensor feedback]. Pset_RobotAccessibility (attached to IfcDoor, IfcSpace) specifies MinClearanceWidth (meters), MinClearanceHeight (meters), AccessibilityStatus {FULL_ACCESS, RESTRICTED, BLOCKED, PENDING_VALIDATION}, ValidationMethod {SENSOR_BASED, GEOMETRIC_ANALYSIS, MANUAL}, and LastValidated (timestamp). Pset_RobotCoordination (attached to IfcSpace) establishes MaxSimultaneousRobots (integer), MinInterRobotSeparation (meters), CoordinationProtocol {YIELD, PRIORITY_PASS, SPEED_REDUCTION}, and AppliesTo {ALL_ROBOTS, SPECIFIC_TYPES}.

Following semantic enrichment, building data is exported in dual complementary formats: (1) geometric representations as FBX format preserving 3D mesh geometry, hierarchical structure, and material assignments for Unity import, and (2) semantic data as CSV tabular format containing element GUIDs, POI classifications, behavioural role codes, and property set values for runtime mapping within the VR simulation environment. This dual-export strategy ensures that both geometric fidelity and semantic richness are transferred to the VR platform while maintaining clear data provenance linking virtual building elements to their authoritative IFC representations.

To validate the formal mapping framework, this research applies the enrichment methodology to a representative subset of IFC entity types covering structural elements, opening elements, and spatial zones essential for indoor robot navigation. Table 3 presents POI role assignments for the demonstration entities, illustrating systematic application of the formal mapping rules defined in Table 1. The framework is not limited to these six entities; rather, the function-based mapping rules (R1–R5) generalize to any IFC-compliant building model. Additional entity types, such as IfcColumn, IfcRamp, IfcStair, IfcFurniture, and IfcEquipment, can be enriched through identical rule application based on their functional characteristics. Also, Figure 2 presents an example of a visual POI assigned role to IFC classes.

Table 3 POI role assignments for demonstration

IFC Entity	Function	POI Category	Roles
IfcWall (exterior)	Building boundary	Structural POI	A-1, A-3
IfcWall (corridor)	Navigation boundary	Structural POI	A-1, A-3
IfcDoor	Space transition	Structural POI	A-2; cond. B-2
IfcSpace (corridor)	Shared navigation	Functional POI	A-3, C-1
IfcSpace (room)	Task destination	Functional POI	Destination
IfcSlab (floor)	Surface reference	Structural POI	A-3



(a)

(b)

Figure 2. Test Environment Configuration

2.3 POI-Driven Navigation Logic

The POI-enriched IFC data informs robot navigation through a systematic decision process that queries semantic properties at runtime and translates POI role assignments into behavioural responses. Algorithm 1 presents the navigation logic executed when robot sensors detect building elements within operational range. The algorithm demonstrates how POI semantic enrichment enables context-aware navigation decisions: sensor detections are matched to IFC entities via GUIDs, POI roles determine behavioural responses, and conditional roles are dynamically activated based on runtime spatial assessments.

Algorithm 1: POI-Driven Navigation Decision

Input: Robot position P, Sensor readings S, POI database D

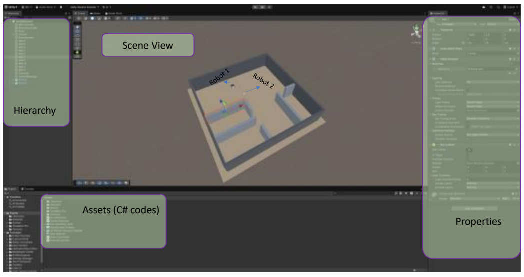
Output: Navigation action A

```
Begin
    FOR each element E detected by sensors S:
        GUID = GetElementIdentifier(E)
        POI = QueryDatabase(D, GUID)
        roles = POI.PrimaryRoles UNION POI.ConditionalRoles
        IF A-1 (Avoidance) IN roles THEN
            MaintainClearance(E, minimum_distance)
        END IF
        IF A-2 (Transition) IN roles THEN
            clearance = POI.Pset_RobotAccessibility.MinClearanceWidth
            IF SensorClearance(E) < clearance THEN
                AddConditionalRole (E, B-2)
                SetAccessibilityStatus(E, RESTRICTED)
            ELSE
                SetAccessibilityStatus(E, FULL_ACCESS)
            END IF
        END IF
        IF C-1 (Multi-Robot Zone) IN roles THEN
            separation = POI.Pset_RobotCoordination.MinInterRobotSeparation
            IF DetectRobot(S) AND RobotDistance() < separation THEN
                AddConditionalRole (E, C-2)
                ExecuteProtocol(SPEED_REDUCTION)
            END IF
        END IF
        IF A-3 (Path Guidance) IN roles THEN
            UseAsTrajectoryReference(E)
        END IF
    END FOR
    RETURN ComputeNavigationAction(roles)
        End
```

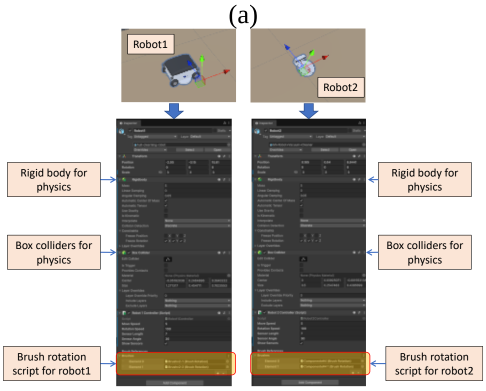
2.4 Robot Navigation Platform in Unity VR

The robot navigation platform is implemented using Unity 6, a real-time 3D development engine providing physics simulation, sensor emulation, and interactive visualization capabilities. The FBX geometric model is imported into Unity, reconstructing the building environment with accurate spatial relationships and dimensional properties. Each imported building element is configured with physics components, including colliders for collision detection and rigid body properties where dynamic interactions are required. The CSV semantic data file is parsed within Unity using a custom POI Manager script that establishes mappings between Unity GameObjects (identified by name) and their associated IFC entities (identified by GUID). Each building element GameObject receives a POI Component attachment encoding its POI category, type, behavioural roles, and robot-navigation parameters extracted from the CSV data, enabling runtime queries by robot navigation algorithms to determine appropriate behavioural responses.

Two robot platforms are instantiated as autonomous agents within the Unity environment, each equipped with physics-based locomotion systems (mass: 5 kg, movement speed: 5.0 m/s, rotation speed: 100°/s), five-directional sensor arrays, and rotating brush mechanisms (500°/s). The sensor configuration projects rays at angular offsets of 0° (forward-center), ±30° (forward-angled), and ±90° (lateral) with 5.0m detection range, providing real-time visual feedback through color-coded visualization: Robot1 displays green (clear) and red (obstacle detected) sensor rays, while Robot2 displays green and blue rays for multi-robot visual distinction during simultaneous operation. Independent control schemes enable concurrent operation: Robot1 via arrow keys and Robot2 via WASD keys, with graphical interface buttons providing brush activation toggles. A dual-camera system (top-down and perspective views) switchable via keyboard input enables comprehensive observation of navigation scenarios from multiple vantage points. Figure 3 presents the Unity platform interface and robot physics configuration.



(a)



(b)

Figure 3. (a) Interface of the platform, and (b) example of physics properties assignment into the robot.

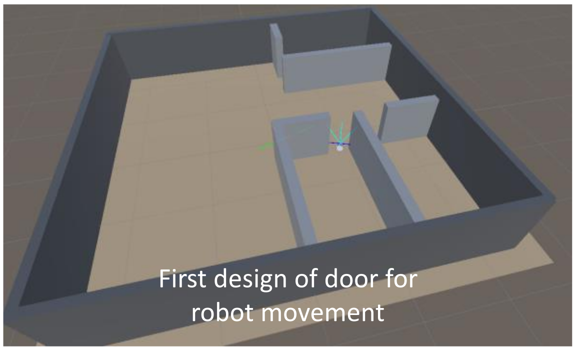
3 Validation Scenarios

The POI-enriched framework is validated through three sensor-based navigation scenarios demonstrating how semantic building information directly informs robot behavioural responses.

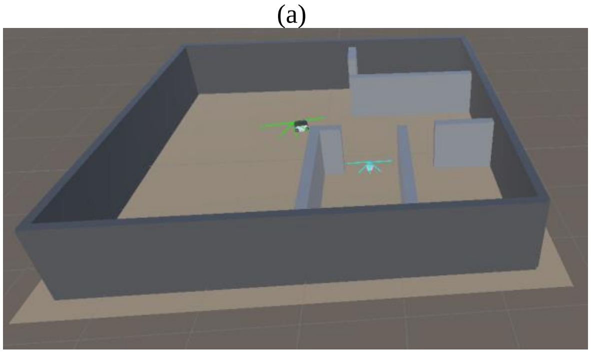
3.1 POI-Driven Doorway Accessibility Validation

The interior doorway connecting the central corridor to Room 2 is initially designed at 0.80m width, meeting standard architectural requirements for human passage but not validated for robot clearance constraints. The doorway's IFC entity (IfcDoor) is enriched with Pset_POI_Classification, assigning Structural POI category, Transition Element type, and A-2 (Transition/Passage) primary role, while Pset_RobotAccessibility specifies MinClearanceWidth = 0.90m as the robot platform's minimum safe passage requirement. As Robot2 approaches the doorway from the corridor at 5.0 m/s navigation speed, the five-directional sensor array evaluates spatial clearances against POI-defined constraints. The forward-center sensor (0°) maintains green status throughout approach, detecting unobstructed passage through the doorway centerline. However, both forward-angled sensors (+30° and −30°) transition to blue at 1.8m approach distance, detecting door frame edges within their angular projection cones at ranges below the 5.0m detection threshold. This sensor pattern indicates that while direct centerline passage is geometrically feasible, approach trajectories at ±30° encounter insufficient clearance margins. The simultaneous blue status of angled sensors triggers conditional POI role addition: the doorway classification updates from A-2 (Transition/Passage) to A-2 + B-2 (Transition with Narrow/Congested constraint), and Pset_RobotAccessibility.AccessibilityStatus property is set to restricted.

Based on sensor feedback indicating inadequate clearance at angled approach vectors, the building geometry is iteratively refined within the BIM authoring environment. The doorway width is increased from 0.80m to 1.20m (50% expansion), the IFC model is re-exported, and POI enrichment is reapplied with updated dimensional properties automatically reflected in Pset_RobotAccessibility parameters. Robot2's subsequent approach to the modified doorway produces distinctly different sensor responses: all forward sensors (0°, ±30°) now maintain GREEN status throughout the approach, indicating that door frame edges remain outside the ±30° detection cones even at proximity. The POI conditional role reverts to standard A-2 (Transition/Passage), and AccessibilityStatus updates to FULL_ACCESS. Figure 4 presents comparative sensor visualizations showing before-modification (BLUE angled sensors) and after-modification (GREEN angled sensors) states.



(a)

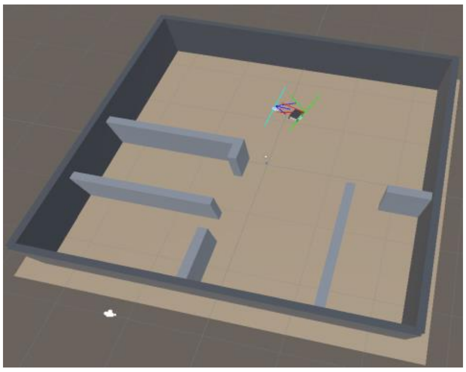


(b)

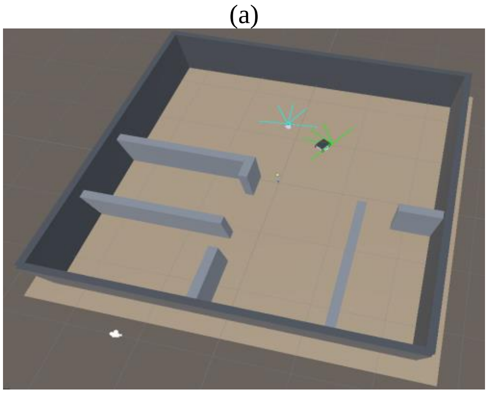
Figure 4. Doorway Validation Comparison

3.2 Multi-Robot Safe Separation Validation

Multi-robot coordination capabilities are validated through scenarios where both Robot1 and Robot2 navigate shared corridor spaces classified as Functional POI with C-1 (Multi-Robot Zone) role assignments. The central corridor IfcSpace entity is enriched with Pset_RobotCoordination specifying MaxSimultaneousRobots, MinInterRobotSeparation of 2.0m between them, and CoordinationProtocol set to SPEED_REDUCTION. As the two robots approach each other from opposite corridor ends, their sensor arrays continuously monitor inter-robot distances. When the separation distance decreases below the 2.0m threshold defined in the POI metadata, both robots trigger C-2 (Robot-Robot Interaction) conditional role activation and execute coordinated speed reduction from 5.0 m/s to 2.5 m/s (50% reduction), maintaining safe operational spacing while allowing continued navigation through the shared zone. Visual sensor feedback differentiates robot detection through color coding: Robot1 displays red rays when detecting Robot2, while Robot2 displays blue rays, enabling clear visualization of proximity detection and coordination protocol execution. This validation demonstrates how the Pset_RobotCoordination property sets embedded in IfcSpace entities directly govern multi-agent behavioural responses without requiring external coordination systems.



(a)



(b)

Figure 5. Multi-robot safe separation validation

3.3 Corridor Navigation Using Wall-Following Behaviour

Robot1 navigates along the central corridor by maintaining alignment parallel to the left wall boundary, classified as a Structural POI with A-3 (Path Guidance) role. The lateral sensor (+90°) continuously monitors the wall at 0.5m standoff distance, providing trajectory reference for corridor navigation. The wall POI classification enables the robot to use the corridor boundary as a navigation reference point, maintaining centered trajectory while moving toward the target room. This scenario validates how POI-enriched building elements provide spatial guidance for autonomous navigation beyond simple obstacle avoidance.

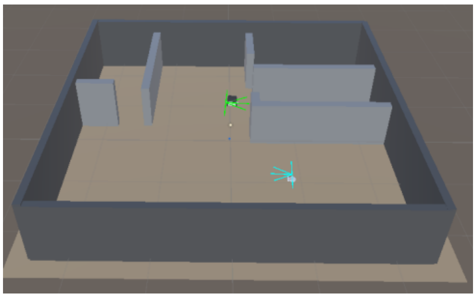


Figure 6. Corridor Navigation Using Wall-Following Behaviour

4 Discussion and limitations

The integration of POI classifications into IFC building models extends BIM utility beyond architectural documentation into autonomous robot operations. Traditional IFC entities provide geometric and material specifications adequate for construction management, yet lack behavioural semantics necessary for robot navigation algorithms to make context-aware decisions. The POI enrichment methodology transforms passive building descriptions into active navigation frameworks by augmenting standard IFC property sets with robot-operational intelligence accessible at runtime.

POI semantic enrichment operates at three levels: classification (Structural, Functional, Object-based categories organize elements by navigation role), behavioural (role codes encode specific responses such as deceleration near narrow passages or coordination in shared corridors), and validation (POI-defined thresholds provide quantitative criteria for automated accessibility assessment). The doorway validation scenario demonstrates proactive design optimization absent from conventional BIM workflows, rather than discovering accessibility constraints post-construction, the POI-driven simulation identified insufficient clearances during design, enabling cost-effective geometric modifications validated through iterative virtual testing.

The current implementation exhibits four primary limitations. First, the validation environment represents simplified geometry that does not capture real-world facility complexity, including furniture clutter, irregular layouts, and environmental variability. Second, validation relies exclusively on virtual Unity simulation rather than physical robot deployments in real buildings. Third, manual semantic enrichment becomes labour-intensive for large-scale projects; automated POI assignment algorithms would significantly improve scalability. Fourth, the POI role taxonomy has been validated only for a subset of role codes through demonstration scenarios, with comprehensive empirical validation across all role groups remaining necessary for full framework validation.

5 Conclusion

This research presents a framework for semantic POI-driven robot navigation through systematic IFC building model enrichment. By augmenting standard BIM entities with robot-oriented POI classifications and behavioural role assignments via three custom property sets (Pset_POI_Classification, Pset_RobotAccessibility, Pset_RobotCoordination), the framework transforms conventional architectural models into semantically rich spatial representations that directly support autonomous navigation while maintaining full compatibility with existing BIM workflows. Unity VR-based validation confirms that POI-enriched building models provide actionable guidance for robot systems, with the doorway accessibility case study demonstrating sensor-based validation that identified and corrected spatial constraints during virtual design iteration before physical construction.

The framework's contribution extends beyond technical implementation to establish conceptual foundations for robot-compatible building design. By formalizing robot-operational requirements as standardized IFC metadata, the research provides a pathway toward automated accessibility assessment tools and building standards that explicitly accommodate autonomous robotic systems alongside human occupants. Future research directions include validation in complex real-world environments, integration with physical robot platforms, development of automated POI assignment algorithms, and engagement with BuildingSMART International to establish formal POI classification standards within the IFC specification.

References

[1] B.M. Tehrani, S. BuHamdan, A. Alwisy, Robotics in industrialized construction: an activity-based ranking system for assembly manufacturing tasks, Engineering, Construction and Architectural Management 31 (2024) 2264–2285. https://doi.org/10.1108/ECAM-02-2022-0143.

[2] H. Oyediran, W. Turner, K. Kim, M. Barrows, INTEGRATION OF 4D BIM AND ROBOT TASK PLANNING: CREATION AND FLOW OF CONSTRUCTION-RELATED INFORMATION FOR ACTION-LEVEL SIMULATION OF INDOOR WALL FRAME INSTALLATION, Journal of Information Technology in Construction 30 (2025) 352–374. https://doi.org/10.36680/j.itcon.2025.015.

[3] Z. Ren, H. Jin LEE, M. Shoaib Khan, X. Zeng, T. Yeon KIM, J. In KIM, Science Mapping for Advances in Integration of Domain Knowledge into Autonomous Construction Robots, (n.d.). https://doi.org/10.6106/ICCEPM.2025.0504.

[4] P. Boguslawski, S. Zlatanova, D. Gotlib, M. Wyszomirski, M. Gnat, P. Grzempowski, 3D building interior modelling for navigation in emergency response applications, International Journal of Applied Earth Observation and Geoinformation 114 (2022). https://doi.org/10.1016/j.jag.2022.103066.

[5] Y.H. Lin, Y.S. Liu, G. Gao, X.G. Han, C.Y. Lai, M. Gu, The IFC-based path planning for 3D indoor spaces, Advanced Engineering Informatics 27 (2013) 189–205. https://doi.org/10.1016/j.aei.2012.10.001.

[6] Q. Chen, J. Chen, W. Huang, Pathfinding method for an indoor drone based on a BIM-semantic model, Advanced Engineering Informatics 53 (2022). https://doi.org/10.1016/j.aei.2022.101686.

[7] X. Zhou, Q. Xie, M. Guo, J. Zhao, J. Wang, Accurate and Efficient Indoor Pathfinding Based on Building Information Modeling Data, IEEE Trans. Industr. Inform. 16 (2020) 7459–7468. https://doi.org/10.1109/TII.2020.2974252.

[8] A. Khalili, D.K.H. Chua, IFC-Based Graph Data Model for Topological Queries on Building Elements, Journal of Computing in Civil Engineering 29 (2015). https://doi.org/10.1061/(asce)cp.1943-5487.0000331.

[9] M.S. Khan, M. Adil, A. Khan, Assessment of structural design capability of building information modeling (BIM) in building industry of Pakistan, JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES 14 (2019) 385–401. https://doi.org/10.26782/jmcms.2019.04.00030.

[10] M.S. Khan, H.S. Cho, J. Seo, Voxel-Based Digital Twin Framework for Earthwork Construction, Applied Sciences 15 (2025) 7899. https://doi.org/https://doi.org/10.3390/app15147899.

[11] M.S. Khan, M. Khan, M. Bughio, B.D. Talpur, I.S. Kim, J. Seo, An Integrated HBIM Framework for the Management of Heritage Buildings, Buildings 12 (2022). https://doi.org/10.3390/buildings12070964.

[12] J. Zhu, M.O. Wong, N. Nisbet, J. Xu, T. Kelly, S. Zlatanova, I. Brilakis, Semantics-based connectivity graph for indoor pathfinding powered by IFC-Graph, Autom. Constr. 171 (2025). https://doi.org/10.1016/j.autcon.2025.106019.

[13] R. Sacks, L. Ma, R. Yosef, A. Borrmann, S. Daum, U. Kattel, Semantic Enrichment for Building Information Modeling: Procedure for Compiling Inference Rules and Operators for Complex Geometry, Journal of Computing in Civil Engineering 31 (2017) 1–12. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000705.