ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XII-4/W1-2026
21st International 3D GeoInfo Conference 2026, 28–29 September 2026, Sofia, Bulgaria

Extraction of Interior and Exterior IFC Building Surfaces by Visibility-Based Mapping for Point-Cloud-to-BIM Comparison

Radmir Gesler[1], Rico Richter[2], Ute Wagner[1], Margitta Pries[1]

[1] BHT, Berliner Hochschule für Technik, Department II. Mathematics - Physics - Chemistry,
Luxemburger Straße 10, 13353 Berlin (radmir.gesler, ute.wagner, pries)@bht-berlin.de
[2] Point Cloud Technology GmbH, August-Bebel-Str. 26-53, 14482 Potsdam, rico.richter@pointcloudtechnology.com

Keywords: BIM, IFC, mesh extraction, IfcOpenShell, point cloud, digital twin

Abstract

This work presents an element-consistent method for extracting interior and exterior surfaces from Industry Foundation Classes (IFC) building models. The approach addresses the geometric and semantic mismatch between terrestrial laser scans representing the *as-is* state and IFC models describing the *as-designed* state. Reliable comparison requires geometrically faithful reference surfaces whose patches remain traceable to their source Building Information Modeling (BIM) elements. The proposed automated pipeline preserves the IFC Level of Development (LOD) without element-wise simplification and maintains each element's GUID throughout processing. Interior room surfaces and exterior shells are formulated within a unified visibility-based framework in which first-hit triangles are identified directly on IFC-derived meshes by ray tracing. The workflow comprises IFC tessellation, storey-wise room-footprint detection, prism-based interior sampling, first-hit triangle aggregation, element-preserving clipping, and analogous exterior visibility-shell extraction. The Python prototype uses IfcOpenShell and Open3D and processes complex building models without manual intervention. The resulting element-aware meshes provide reference geometry for Scan-to-BIM registration, geometric deviation analysis, and digital twin validation.

1. Introduction

The use of BIM and 3D sensing technologies has become an integral part of digital construction workflows. In particular, comparing dense terrestrial laser scans representing the *as-is* state with IFC-based BIM models describing the *as-designed* state is important for quality assessment, progress monitoring, and digital twin validation. This requires not only geometric alignment, but also a consistent mapping between observed *as-is* surfaces and the corresponding semantic building elements in the IFC model.

This work is deliberately positioned on the BIM side of the Scan-to-BIM workflow. It does not address the complete registration problem between point clouds and BIM geometry. Instead, it constructs geometrically detailed and semantically traceable IFC reference surfaces required before registration, distance computation, and element-level deviation reporting can be performed reliably.

IFC is the only widely adopted open and vendor-neutral standard for BIM data exchange (Gerbino et al., 2021). Contemporary IFC models encode geometry, semantic information, and element relationships (Kolbe and Donaubauer, 2021). However, their geometric content varies considerably with modelling practices, authoring tools, and project requirements. Models may also contain auxiliary objects, furnishings, or contextual geometry that are irrelevant to geometric analysis (Noardo et al., 2021b). Many downstream workflows therefore require the extraction of precise, application-specific, and semantically traceable surfaces directly from IFC.

The central challenge is not merely geometric filtering, but constructing element-consistent interior and exterior surface representations that preserve the original LOD and remain unambiguously linked to their source IFC entities. Such representations provide the basis for Scan-to-BIM comparison, deviation analysis, and digital twin validation.

We present a unified visibility-driven framework operating directly on IFC-derived triangle meshes. Interior room surfaces and the exterior shell are formulated as visibility problems. Sampling points are generated inside room volumes and outside the building envelope, from which rays are cast and first-hit triangles are accumulated. The resulting submeshes retain the GUIDs and semantic attributes of their source elements, yielding element-aware reference geometry without element-wise simplification.

The main contributions are:

1. A unified visibility-driven framework for extracting interior and exterior IFC surfaces directly from triangle meshes.

2. An element-consistent mapping that preserves IFC GUIDs throughout the extraction process and enables traceable geometric analysis.

3. A formal room-wise sampling and visibility formulation, including grid-based interior viewpoints and first-hit mapping from rays to IFC element triangles.

2. Related Work

Research on indoor representations derived from IFC models can be grouped into three directions: (i) generation of space boundaries for building simulation, (ii) conversion to indoor network models such as IndoorGML[1], and (iii) semantic

[1] IndoorGML: Open Geospatial Consortium (OGC) Standard for Indoor Spatial Information

indoor-space extraction. These approaches primarily emphasize abstraction and semantic reasoning rather than high-fidelity surface extraction.

For simulation, Fichter et al. (2021) generated IFC-based boundaries for CFD[2] and BEPS[3], while Ying and Lee (2021) and Lilis et al. (2017) reconstructed second-level space boundaries from IFC geometry. Although effective for simulation preprocessing, these methods replace the original IFC mesh geometry with boundary abstractions.

IFC-to-network methods generate IndoorGML or Building Configuration Models for navigation, spatial analysis, and BIM–GIS interoperability (Teo and Yu, 2017; Jia et al., 2025; Diakite et al., 2022). Their focus is topology, connectivity, and semantic enrichment rather than exact surface preservation.

Recent work also combines indoor-space extraction with semantic modeling. For example, Zhao et al. (2025) constructs topological graphs of interior subspaces for graph-neural-network-based classification.

Consequently, existing methods generally do not preserve an element-consistent mapping between visible building surfaces and their originating IFC triangle meshes. Geometric abstraction, boundary reconstruction, or network conversion breaks this correspondence. For Scan-to-BIM registration, however, reference surfaces must remain assignable to IFC elements so that deviations can be reported reliably for walls, slabs, windows, and other BIM entities.

In contrast, the present work extracts element-consistent surfaces directly from IFC meshes without geometric simplification. The proposed method preserves the original LOD while maintaining a one-to-one linkage between extracted surface patches and their originating IFC entities.

3. Data Assumptions and Problem Setting

The proposed framework assumes that the IFC model $\mathcal{M}$ follows the standard hierarchical spatial decomposition shown in Figure 1. Physical building elements e are assigned to their corresponding `IfcBuildingStorey` through `IfcRelContainedInSpatialStructure`, enabling storey-wise processing. Selected elements must be tessellable

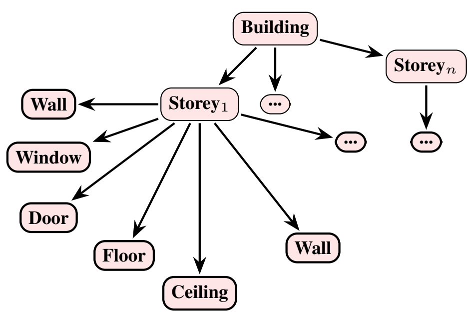


Figure 1. An image mapping of hierarchical IFC building model.

into valid manifold triangle meshes using IfcOpenShell[4] and provide a stable GUID. Elements with missing geometry, degenerate meshes, or inconsistent placements are discarded during preprocessing.

[2] Computational Fluid Dynamics
[3] Building Energy Performance Simulation
[4] IfcOpenShell: the open-source IFC toolkit and geometry engine.

Since `IfcSpace` entities are frequently incomplete or absent in real-world IFC models (Lilis et al., 2021; Noardo et al., 2021a), room candidates are derived directly from physical building elements. The model is tessellated into per-element meshes $\{M_e\}$ for selected IFC elements, while non-physical entities, furniture, and contextual objects may be excluded by type or name filters.

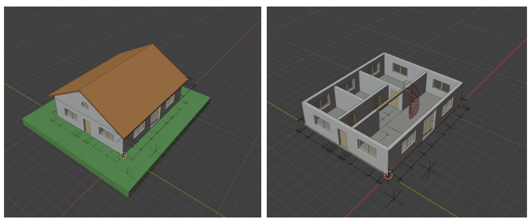
The method preserves the geometric LOD while separating geometric availability from task relevance. Element participation in the interior or exterior visibility scene is controlled by IFC class and name filters, orientation tests, and spatial intersection tests. For example, wall elements are retained for wall-footprint generation, whereas predominantly horizontal floor-, ceiling-, and covering-like elements are excluded.

Within this setting, two complementary extraction tasks are addressed:

1. **Interior-mesh-extraction:** Given a storey $E \subset \mathcal{M}$ and its associated elements $\{e\}$, determine the set of interior surface triangles visible from within room volumes.
2. **Exterior-mesh-extraction:** Given the entire model $\mathcal{M}$, extract per-element triangle subsets visible from the exterior, thereby approximating the external building visibility shell.

4. Method

The methodology is demonstrated using the publicly available AC20-FZK-Haus IFC model (Figure 2). The objective is to extract element-consistent interior and exterior IFC surfaces for traceable comparison with terrestrial laser scans representing the building's *as-is* state.



(a) Whole modell

(b) The ground floor of the model from Figure 2a

Figure 2. An IFC model of a house for demonstration of the methodology. The images were created with Blender software.

4.1 Interior-Mesh-Extraction

We first address the **Indoor-Mesh-Extraction** problem. Let a storey-specific set of physical building elements be defined as

$$E = \{e_i \in \mathcal{M} \mid i \in \mathcal{I}_E\}, \;\; \mathcal{I}_E \text{ set with element ID's.} \tag{1}$$

For each geometric element $e \in E$, a triangle mesh M_e is generated from its IFC representation. The storey-specific element set E, defined in (1), is then processed using Pipeline 1, as described below.

Lines 1–3 perform the geometric preprocessing: walls are filtered, projected to planar footprints, and their closed inner rings are identified as candidate room contours (Figure 3).

Pipeline 1 Indoor geometry extraction at storey level E using wall projection and ray tracing

Require: E, h (point grid size), n (number of rays per point)
Ensure: For each prism pr: set of internally visible triangle faces $M_{\text{in}}(pr)$

1: $E_W \leftarrow$ ExtractWallMeshes(E)
2: $P \leftarrow$ WallsFootprint2D(E_W)
3: $P_{\text{inr}} \leftarrow$ EnclosedRoomsFromWalls(P)
4: $Pr \leftarrow$ RoomsPolysToPrisms(P_{inr})
5: $\mathcal{Q}_\partial \leftarrow$ DistributePointsInPrisms(Pr, h)
6: $\mathcal{R} \leftarrow$ BuildRaysForRooms$(\mathcal{Q}_\partial, n)$
7: $\mathcal{T} \leftarrow$ ExtractInternalMesh(E, pr)
8: **return** $\mathcal{T}(e_\Delta)$

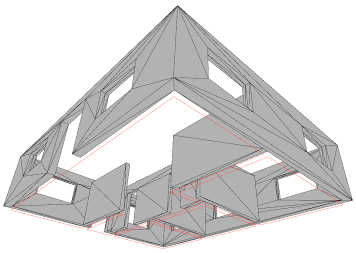


Figure 3. The result of detecting the interior space counters.

Line 4 constructs sampling points within each detected room prism. Let $P_r \subset \mathbb{R}^2$ denote the room footprint from Line 3, with (u, v) spanning the footprint plane and w the extrusion direction. The prism is defined as

$$pr_r = \left\{x \in \mathbb{R}^3 \mid (x_u, x_v) \in P_r,\ w_{\min} \leq x_w \leq w_{\max}\right\} \quad (2)$$

where $(w_{\min}, w_{\max})$ denotes the estimated wall-height range.

In Line 5, a regular grid with spacing h is constructed over the bounding box of P_r, retaining only pairs with $(u_i, v_j) \in P_r$. The lower sampling points are

$$\mathcal{Q}_0 = \{(u_i, v_j, w^0_{min}) \mid\ w^0_{min} = w_{\min} + \beta,\ (u_i, v_j) \in P_r\}, \quad (3)$$

where β is a small bottom margin. To adapt the sampling volume to sloped or locally varying ceilings, an upward point $h_{ij} = q^0_{ij} + t^*_{ij} e_w - \alpha e_w,\ t^* \geq 0$ is cast from every $q^0_{ij} \in \mathcal{Q}_0$ against the room-related IFC meshes. With the first positive hit on t^* defines a local upper point, where α is a top margin preventing sample points from lying exactly on the ceiling mesh. The volumetric sampling set is then obtained by vertical interpolation between q^0_{ij} and h_{ij} gives the point set $\mathcal{Q}_{\text{vol}}$. Subsequently, points falling inside an oriented bounding box of a room-intersecting IFC element are removed. With B^ρ_e denoting the oriented bounding box of element e expanded by a tolerance ρ, this free-space point set is

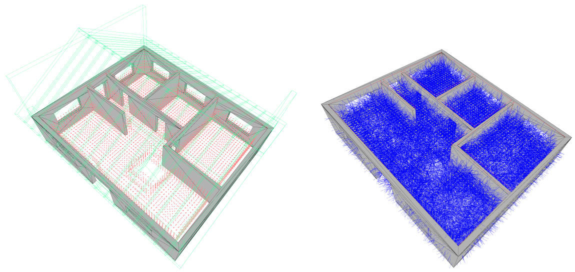
$$\mathcal{Q}_{\text{free}} = (\mathcal{Q}_{\text{vol}} \cap pr_r) \setminus \bigcup_{e \in E_{pr_r}} B^\rho_e. \quad (4)$$

Finally, only the outer grid layer is kept by using the six-neighbourhood ($\mathcal{N}_6$) in the grid. A point q is kept as a light-source point if at least one of its direct neighbours is missing:

$$\mathcal{Q}_\partial = \{q \in \mathcal{Q}_{\text{free}} \mid \exists n \in \mathcal{N}_6(\kappa(q)):\ n \notin \kappa(\mathcal{Q}_{\text{free}})\}. \quad (5)$$

Thus, the ray origins are concentrated close to the boundary of the free room volume.

Line 6 completes the volumetric sampling procedure: room polygons are extruded to the estimated wall height (Line 4), boundary sampling points $\mathcal{Q}^r_\partial$ are generated within each prism (Line 5), and n approximately uniform directions on $\mathbb{S}^2$ are assigned to each point (Line 6). Figures 4a and 4b show the resulting points and rays.



(a) Regular grid $\mathcal{Q}_\partial$ of light points in the spatial prisms.

(b) 1024 uniformly distributed rays from a sphere point (for illustration, only 10 rays per point and each $1m$ of length.

Figure 4. Visualization of the sampling results showing the grid distribution of light points inside the room prisms and the uniform ray generation for each sampling point.

Note: Boundary elements such as slabs, roofs, coverings, or ceilings may intersect a room despite being assigned to another storey. The implementation therefore selects all physical elements whose bounding boxes intersect the room-prism bounding box,

$$E_{pr_r} = \{\, e \in E_{\text{phys}} \mid \text{AABB}(M_e) \cap \text{AABB}(pr_r) \neq \emptyset \,\}. \quad (6)$$

This conservative selection may include additional geometry but avoids missing boundary elements with incomplete or inconsistent storey assignments.

Before presenting the results of line 7, we briefly introduce the procedure's core idea.

Ray-based visibility extraction of interior surfaces. Let E denote the set of all building elements of a storey and let each element $e \in E$ be represented by a triangle mesh $M_e = \bigcup_{\Delta \in \mathcal{F}(e)} \Delta$ where $\mathcal{F}(e)$ denotes the set of triangular faces of e. Since the interior room prism $pr \subset \mathbb{R}^3$ is derived from the wall footprint projection, we can build the subset of elements intersecting the prism pr as $E_{pr} = \{\, e \in E \mid M_e \cap pr \neq \emptyset \,\}$. In the implemented pipeline, this definition is evaluated in the conservative sense given in (6). Within the interior of the prism, a finite set of sampling points $\mathcal{Q}_\partial = \{q_j\}^k_{j=1} \subset \text{int}(pr)$ is distributed. And for each point q_j a finite set of normalized direction vectors $\mathcal{R}(q_j) = \{\mathbf{d}_\ell\}^n_{\ell=1} \subset \mathbb{S}^2$ is generated, where $\mathbb{S}^2$ denotes the unit sphere. For each direction $\mathbf{d}_\ell$ a ray can be defined as $r_{j,\ell}(t) = q_j + t\,\mathbf{d}_\ell,\ t \geq 0$. For some triangles $\Delta \subset M_e$ with its vertices $v_1, v_2, v_3 \in \mathbb{R}^3$, every point inside can be represented using barycentric coordinates $x = v_1 + u(v_2 - v_1) + v(v_3 - v_1)$ with $u \geq 0,\ v \geq 0,\ u + v \leq 1$. Now for every ray $r_{j,\ell}$ the first intersection with a triangle is conceptually determined by

$$q_j + t\,\mathbf{d}_\ell = v_1 + u(v_2 - v_1) + v(v_3 - v_1). \quad (7)$$

https://doi.org/10.5194/isprs-annals-XII-4-W1-2026-153-2026

And finally the intersection can be denoted as a parameter

$$t^*_{j,\ell} = \min\{t > 0 \mid r_{j,\ell}(t) \in \Delta \subset M_e \text{ for some } e \in E_{pr}\}. \tag{8}$$

If such an intersection exists, the corresponding triangle $\Delta_{j,\ell}$ is called the *first-hit triangle* of the ray. Each triangle is uniquely associated with the element e to which it belongs. The set of triangles detected for an element e is therefore

$$e_\Delta = \{\Delta_{j,\ell} \mid \Delta_{j,\ell} \subset M_e \text{ and } t^*_{j,\ell} < \infty\}. \tag{9}$$

Figure 5 illustrates the fundamental principle of the first-hit ray-tracing approach.

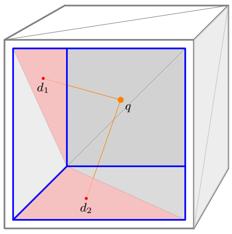


Figure 5. Illustration of the ray-tracing method for extracting the interior mesh from a cubic body containing both inner and outer mesh structures. The light source point q inside the blue prism emits, exemplarily, two rays. These rays intersect the triangles on d_1 and d_2. The intersected triangles are recorded together with the corresponding IFC element identifiers (GUID).

The triangles in e_Δ from (9) form the interior-visible subset of the original element mesh. To ensure consistency with the room geometry, they are clipped to the prism volume:

$$e_\Delta(pr) = \{\Delta \cap pr \mid \Delta \in e_\Delta\}. \tag{10}$$

The complete interior-visible mesh set is

$$M_{\text{in}}(pr) = \{\, e_\Delta(pr) \mid e \in E_{pr}\}. \tag{11}$$

Each submesh retains the identifier of its source IFC element, enabling element-wise analysis and room-level processing.

Algorithm 2 summarizes the complete procedure.

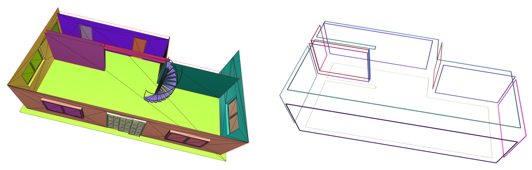
Algorithm 2 Ray-tracing based extraction of interior visible surfaces

Require: Element set E, interior prisms Pr, ray sets $\mathcal{R}(pr)$
Ensure: Interior visible meshes $M_{\text{in}}(pr)$ for each prism

for all $pr \in Pr$ **do**
  $S \leftarrow$ BUILDRAYCASTINGSCENE()
  Insert element meshes M_e with $e \in E_{pr}$ into S
  $A \leftarrow$ CASTRAYS($S, \mathcal{R}(pr)$)
  Extract first-hit triangle identifiers from A
  Build element-wise triangle sets e_Δ
  $e_\Delta \leftarrow$ CLIPWITHPRISM(e_Δ, pr)
  $M_{\text{in}}(pr) \leftarrow \bigcup e_\Delta$
end for
return M_{in}

Applying the pipeline to a storey-level element set E does not necessarily produce a room mesh that exactly matches the room prism or contains only room-bounding element parts. At this stage, candidate meshes are clipped only by the room prism's *axis-aligned bounding box*, not by the prism itself. This coarse *preselection* retains potentially relevant geometry while avoiding costly operations on the full storey mesh.

Consequently, the preselected mesh may contain geometry outside the prism and may not follow its planar boundaries, as shown in Figure 6a.



(a) Raw extracted spatial room meshes. (b) Decomposed room prism in its faces

Figure 6. A representative result after application of Algorithm 2 on the storey data from Figure 2b (left) and its decomposed room prism (right).

Therefore, we introduce a *face-consistent cutting* procedure that refines each preselected element by clipping it against the set of room prism faces.

Element-wise mesh restriction via planar face assignment. After visibility-based extraction, element meshes are clipped to the room boundaries. For each mesh M_e, a representative plane is estimated from sampled points using principal component analysis (PCA). Let $P = \{p_i\}_{i=1}^N$ denote the sampled point set of M_e. The centroid is defined as $c_e = \frac{1}{N}\sum_{i=1}^{N} p_i$. The covariance matrix $\Sigma = \frac{1}{N}\sum_{i=1}^{N}(p_i - c_e)(p_i - c_e)^\top$ is then spectrally decomposed. The eigenvectors define a local orthonormal frame (n_e, u_e, v_e), where n_e represents the dominant plane normal of the element surface and (u_e, v_e) span the corresponding in-plane coordinate system.

The room prism is decomposed into faces (Figure 6b), each represented by a planar polygonal loop $\Gamma_f \subset \mathbb{R}^3$. Its vertices define a local orthonormal frame (u_f, v_f, n_f) and a reference point $p_{0,f} \in \mathbb{R}^3$, taken as their centroid.

The corresponding face plane is therefore given by

$$\Pi_f := \left\{x \in \mathbb{R}^3 \mid \langle x - p_{0,f}, n_f\rangle = 0\right\}. \tag{12}$$

Projecting Γ_f into the (u_f, v_f) coordinate system and performing planar triangulation yields a set of triangular patches $\{\tau_{f,k}\}_{k=1}^{K_f}$.

An element e is considered a geometric candidate for face f if the angular deviation between the element plane and the face plane and their signed plane distance satisfy $|\langle n_e, n_f\rangle| \geq \cos(\theta_{\text{tol}})$ and $|\langle c_e - p_{0,f}, n_f\rangle| \leq d_{\text{tol}}$.

For each triangular patch $\tau_{f,k}$, we construct a convex clipping prism

$$\mathcal{P}_{f,k} = \bigcap_{i=1}^{5} H_i, \tag{13}$$

as the intersection of five half-spaces: two parallel slab planes and three lateral planes induced by the triangle edges. The slab planes are

$$H_{1,2} = \left\{x \in \mathbb{R}^3 \mid |\langle x - p_{0,f}, n_f \rangle| \le \varepsilon \right\}, \tag{14}$$

defining a thin layer around the face plane, while H_3, H_4, H_5 laterally bound the prism within that plane.

Triangles of M_e whose vertices lie within a threshold distance of Π_f are clipped against $\mathcal{P}_{f,k}$. The resulting polygon fragments are fan-triangulated and accumulated into the restricted interior submesh.

Floor and ceiling polygons Γ_f may be non-convex, for example in L-shaped rooms, recesses, or roof intersections. They are therefore decomposed into convex triangular subdomains

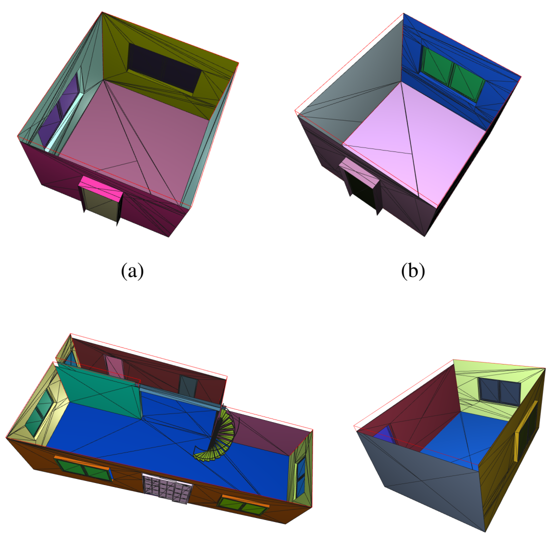
$$\Gamma_f = \bigcup_{k=1}^{K_f} \tau_{f,k}, \qquad \mathrm{int}(\tau_{f,k}) \cap \mathrm{int}(\tau_{f,l}) = \emptyset \quad (k \ne l). \tag{15}$$

Each $\tau_{f,k}$ defines a clipping prism $\mathcal{P}_{f,k}$ as above, and the face-specific restricted mesh is

$$M_e^f = \bigcup_{k=1}^{K_f} \textsc{Clip}(M_e, \mathcal{P}_{f,k}). \tag{16}$$

This avoids replacing concave faces by their convex hull and prevents inclusion of fragments outside the true room footprint.

Finally, we applied the described procedure (Line 7 in Pipeline 1) to the storey data E from the storey-model of 2b and obtained the results shown in Figure 7.



(a) (b) (c) (d)

Figure 7. The extracted internal room meshes from the example in Figure 2, clipped against its room prism faces.

4.2 Exterior-Mesh-Extraction

The extraction of externally visible building surfaces is formulated analogously to the indoor extraction problem, but the sampling points are located outside the building envelope.

Global bounding box. Let $\mathcal{E} \subset \mathcal{M}$ denote the set of selected building elements and let M_e be the corresponding triangle mesh of element $e \in \mathcal{E}$ expressed in a common world coordinate system. A global axis-aligned bounding box is constructed as

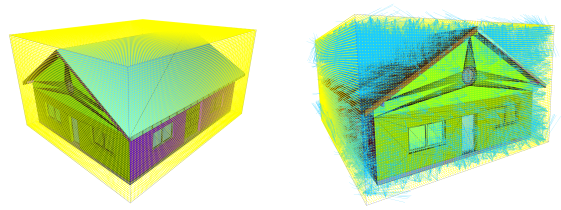
$$\mathcal{B} = \mathrm{AABB}\left(\bigcup_{e \in \mathcal{E}} M_e\right). \tag{17}$$

To guarantee that sampling points lie outside the building geometry, the bounding box is expanded by a padding distance $\delta > 0$, yielding an enlarged sampling domain $\mathcal{B}_\delta$.

Sampling on the bounding box. Sampling points are placed on the six faces of the expanded bounding box $\mathcal{B}_\delta$. A regular grid with spacing h defines the exterior observation points

$$\mathcal{Q} = \{q_j \in \partial\mathcal{B}_\delta \mid j = 1, \dots, N_q\}. \tag{18}$$

For each q_j, directions $\{d_{j,\ell}\}_{\ell=1}^{n_{\mathrm{out}}}$ are sampled within a spherical cap around the inward face normal and normalised. The resulting cone-directed rays explore the model interior while retaining angular diversity (Figure 8).



(a) Light points distributed on the grid of the bounding box $\mathcal{B}$ of the example model 2a.

(b) Illustrative subset of inward-directed rays generated at the sampling points.

Figure 8. Ray preparation for the outer-shell extraction procedure.

Raytracing and triangle aggregation. The extraction of outer shell meshes from $\mathcal{M}$ follows the same principle as *Interior-Mesh-Extraction*. A raycasting scene is constructed from all building meshes $\{M_e\}$. For each ray $r_{j,\ell}(t) \in \mathcal{R}$, the first-hit triangle and its element identifier are recorded. Aggregating the hits yields the externally visible triangle set defining the outer submesh as

$$M_{\mathrm{out}}(e) = \bigcup_{\Delta_i \in \Delta_{\mathrm{out}}(M_e)} \Delta_i. \tag{19}$$

Aggregating these subsets for all elements yields a set of element-consistent exterior surface meshes

$$\mathcal{H}_{\mathrm{out}} = \bigcup_{e \in \mathcal{E}} M_{\mathrm{out}}(e). \tag{20}$$

This representation forms a *visibility shell*, i.e., the set of triangles externally visible from the sampling ensemble. Although not necessarily watertight, it preserves the original IFC element structure and provides a consistent exterior surface representation for downstream tasks such as Scan-to-BIM comparison.

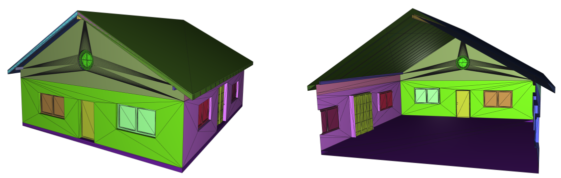
The procedure is summarized in Pipeline 3 and yields an element-consistent representation of externally visible building surfaces.

Pipeline 3 Ray-tracing based extraction of exterior visible surfaces

Require: Building model $\mathcal{M}$, grid size g, number of rays n, cone angle α
Ensure: Per-element mapping $\mathcal{D}$ (GUID $\mapsto$ outer visible submesh $M_{\text{out}}(e)$)

$\mathcal{B} \leftarrow$ BUILDGLOBALAABB($\mathcal{M}$)
$\mathcal{Q} \leftarrow$ DISTRIBUTEPOINTSONGLOBALBB($\mathcal{B}, g$)
$\mathcal{R} \leftarrow$ BUILDRAYSFOROUTERSHELL($\mathcal{Q}, n, \alpha$)
$\mathcal{D} \leftarrow$ RAYCASTANDEXTRACTOUTERSHELL($\mathcal{M}, \mathcal{R}$)
return $\mathcal{D}$

Figure 9 shows the result of applying the exterior shell extraction procedure to the example model in Figure 2a. The extracted geometry represents the externally visible triangle subset while preserving the correspondence between surface patches and their originating IFC elements.



(a) Extracted outer shell of the example model 2a.

(b) Plane cut through the extracted outer shell revealing the interior structure.

Figure 9. Result of the outer-shell extraction process.

4.3 Data structure and traceability

For both extraction pipelines, the resulting geometry is stored in an element-consistent reference structure. Each IFC element stores its GUID together with the optional meshes `indoor_face_mesh` and `outer_face_mesh`, representing the interior- and exterior-visible triangle subsets. This preserves the traceability chain

$$\text{geometry} \rightarrow \text{element} \rightarrow \text{IFC semantics},$$

allowing every extracted triangle to be mapped back to its originating IFC entity. This traceability is essential for downstream Scan-to-BIM tasks, such as reporting geometric deviations between *as-is* point clouds and *as-designed* IFC elements.

4.4 Implementation

The Python prototype combines `IfcOpenShell` for IFC parsing and tessellation with `Open3D` for geometric processing and GPU-accelerated raycasting. Triangle meshes are represented as Open3D tensor meshes.

Because Open3D does not directly map intersected triangles to their source IFC elements, this correspondence is established at the application level. Geometry identifiers and triangle indices returned by the raycasting scene are mapped to the originating meshes and IFC elements, enabling element-wise aggregation of first-hit surface triangles.

Figure 10 shows the prototype architecture. Its core `PC2TwinLibrary` components provide IFC processing, geometric utilities, room extraction, and element decomposition. The current implementation focuses on BIM-side processing; point-cloud integration remains future work.

Element selection is restricted to physical objects from the `IfcElement` branch, while auxiliary geometry may be excluded using type or name filters. A global 4×4 transformation matrix can be applied to all vertices for georeferencing. This is suitable for building-scale models, but may cause distortions in infrastructure- or city-scale datasets.

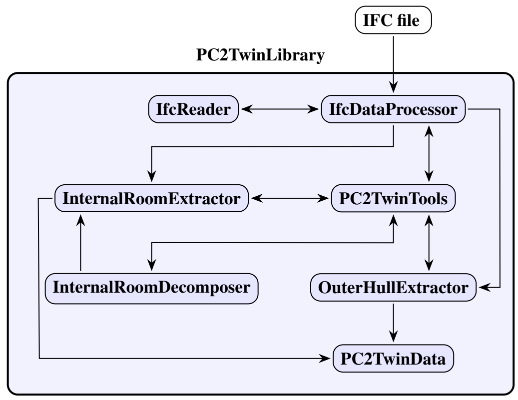
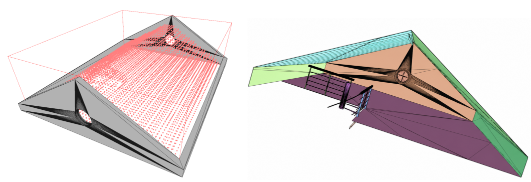


Figure 10. The architecture of the implemented prototype library `PC2TwinLibrary`.

4.5 Discussion and Limitations

The extracted exterior geometry represents a *visibility shell*, i.e., only first-hit triangles from the chosen sampling configuration are retained. Consequently, coarse sampling (grid spacing h or insufficient rays n_{out}) may miss visible surfaces, while unfiltered auxiliary IFC geometry may contaminate the result. Practical applications therefore require appropriate filtering based on IFC type and name patterns.

For interior extraction, the reconstruction quality depends on the modelling style of the IFC dataset and the completeness of the wall footprints. While doors and windows generally do not impair reconstruction if the projected footprint remains topologically closed, the current assumption of vertically extruded room prisms limits the treatment of sloped ceilings, curved walls, and other non-planar geometries. In such cases, insufficient vertical sampling may reduce ray coverage and result in incomplete surface extraction.



(a) Initial sampling configuration with insufficient ceiling margin.

(b) Improved extraction result using configuration 11a.

Figure 11. Influence of light-point grid distribution parameters on the extraction of interior surfaces in non-rectangular rooms.

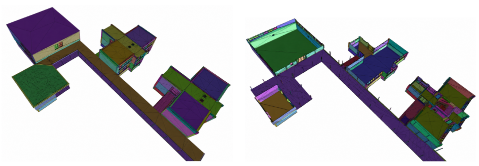
https://doi.org/10.5194/isprs-annals-XII-4-W1-2026-153-2026

Figure 11 demonstrates that increasing the vertical distribution of light points improves ray coverage and the completeness of the extracted interior surfaces.

Future work will extend the sampling strategy using free-space-aware point distributions and investigate alternative visibility computation methods, such as cubemap-based rendering, to improve robustness and computational efficiency.

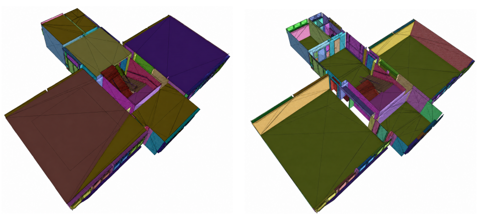
4.6 More complex examples

To evaluate scalability, the prototype was applied to AC20-FZK-Haus and two larger IFC project models using identical sampling parameters. Table 1 summarizes the computational performance, while Figures 12–15 show representative exterior and interior extraction results. As expected, runtime and



(a) Extracted exterior shell of model A. (b) Horizontal section through the extracted exterior shell.

Figure 12. Exterior-shell extraction result for model A, an IFC project model with a file size of 37.6 MB and 515,782 tessellated triangles.



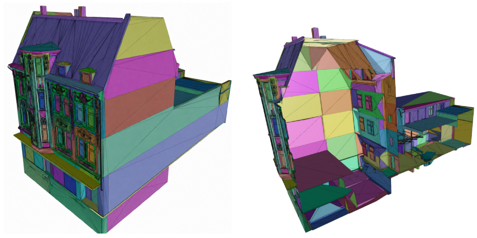
(a) Extracted internal meshes of the ground floor in model A. (b) Horizontal section through the extracted internal meshes.

Figure 13. Room-wise internal mesh extraction for the ground floor of model A, containing eight detected rooms.

memory consumption increase with geometric and sampling complexity. Model A exhibits the highest runtime due to its larger ray count, whereas Model B, despite containing more triangles, is processed faster because fewer rays are generated. This indicates that computational cost depends not only on mesh complexity but also on the sampling configuration and the number of processed rooms and storeys. The extracted meshes demonstrate that the proposed method produces coherent exterior visibility shells and room-wise interior surfaces for both project models while preserving the correspondence to the originating IFC elements, indicating applicability to realistic IFC datasets.

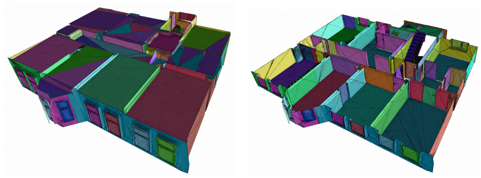
5. Conclusion

This work presented a unified visibility-driven framework for extracting visible interior room surfaces and exterior building surfaces directly from IFC geometry. Both tasks are formulated



(a) Extracted exterior shell of model B. (b) Longitudinal section through the extracted exterior shell.

Figure 14. Exterior-shell extraction result for model B, an IFC project model with a file size of 54.7 MB and 678,708 tessellated triangles.



(a) Extracted internal meshes of the second floor in model B. (b) Horizontal section through the extracted internal meshes.

Figure 15. Room-wise internal mesh extraction for the second floor of model B, containing fourteen detected rooms.

as ray-tracing visibility problems operating on triangle meshes derived from IFC elements.

A key contribution is the preservation of element-level traceability: all extracted surface subsets remain linked to their originating IFC entities through stable GUIDs. The resulting per-element submeshes preserve the original geometric LOD while providing a semantically consistent reference representation for downstream applications such as Scan-to-BIM registration, geometric deviation analysis, and digital twin validation.

Unlike many existing approaches, the proposed method operates directly on IFC meshes without geometric simplification or strict watertightness assumptions, making it applicable to real-world IFC datasets with varying modelling quality. By treating interior and exterior extraction within a common visibility-based framework, it provides a consistent basis for IFC-based geometric processing.

Future work will extend the approach to more general room geometries, including non-rectangular and curved spaces, and integrate a complete point-cloud processing pipeline comprising registration, quantitative deviation analysis, and BCF-based reporting. In addition, the proposed evaluation metrics will be used to benchmark extraction accuracy, runtime, and memory consumption, supporting further optimisation of the sampling strategy and visibility computation.

6. Acknowledgements

The authors gratefully acknowledge pointreef GmbH for providing access to the IFC project models used in this work. We further thank 'Wirtschaftsbetriebe Duisburg' for granting permission to use Model A and 'Baukultur Nordrhein-Westfalen e. V.' for providing Model B. These datasets enabled

Model	IFC size	Triangles	Total rays	Total time	Peak memory	Delta memory
AC20	2.4 MB	32,550	5,015,284	24.71 s	1009.40	327.93 MB
Model A	37.6 MB	515,782	53,751,332	418.51 s	7890.52	2904.77 MB
Model B	54.7 MB	678,708	23,608,304	346.9 s	4361.46	2088.57 MB

Table 1. Overview of computational behaviour for the processed IFC examples.

the evaluation of the proposed extraction workflow on complex real-world building models.

This work was partially funded by the German Federal Ministry for Economic Affairs and Climate Action (BMWK), Germany, under grant FKZ:16KN113528, 16KN113527 (Project 'PointCloud2Twin').

References

Diakite, A. A. et al., 2022. Ifc2indoorgml: An open-source tool for generating indoorgml from ifc. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022, 295–301.

Fichter, E., Richter, V., Frisch, J., van Treeck, C., 2021. Automatic generation of second level space boundary geometry from ifc models. *Proceedings of Building Simulation 2021: 17th Conference of IBPSA*, Building Simulation, 17, IBPSA, Bruges, Belgium, 1083–1090. IBPSA publication page.

Gerbino, S., Cieri, L., Rainieri, C., Fabbrocino, G., 2021. On BIM Interoperability via the IFC Standard: An Assessment from the Structural Engineering and Design Viewpoint. *Applied Sciences*, 11(23). MDPI article.

Jia, Z., Nourian, P., Luscuere, P., Wagenaar, C., 2025. IFC2BCM: A Tool for Generating IndoorGML and Building Configuration Model from IFC. *SoftwareX*, 29, 101975. ScienceDirect page.

Kolbe, T. H., Donaubauer, A., 2021. Semantic 3d city modeling and bim. *Urban Book Series*, Springer, 609–636.

Lilis, G. N., Giannakis, G. I., Rovas, D. V., 2017. Automatic generation of second-level space boundary topology from IFC geometry inputs. *Automation in Construction*, 76, 108–124.

Lilis, G. N., Katsigarakis, K., Rovas, D., 2021. Automatic IFC data enrichment with space geometries for Building Energy Performance Simulations. *Proceedings of Building Simulation 2021: 17th Conference of IBPSA*, 17, 2133–2140. IBPSA publication page.

Noardo, F., Arroyo Ohori, K., Krijnen, T., Stoter, J., 2021a. An Inspection of IFC Models from Practice. *Applied Sciences*, 11(5). MDPI article.

Noardo, F., Krijnen, T., Arroyo Ohori, K., Biljecki, F., Ellul, C., Harrie, L., Eriksson, H., Polia, L., Salheb, N., Tauscher, H., van Liempt, J., Goerne, H., Hintz, D., Kaiser, T., Leoni, C., Warchol, A., Stoter, J., 2021b. Reference study of IFC software support: The GeoBIM benchmark 2019—Part I. *Transactions in GIS*, 25(2), 805–841. Publisher page.

Teo, T.-A., Yu, S.-C., 2017. THE EXTRACTION OF INDOOR BUILDING INFORMATION FROM BIM TO OGC INDOORGML. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W2, 167–170. Article page.

Ying, H., Lee, S., 2021. Generating second-level space boundaries from large-scale IFC-compliant building information models using multiple geometry representations. *Automation in Construction*, 126, 103659. ScienceDirect page.

Zhao, Z., Chen, W., Guo, Z., Jiang, W., 2025. Ifc-based indoor space extraction with topological graph modeling. J. Cheng, Y. Yantao (eds), *Proceedings of The Sixth International Conference on Civil and Building Engineering Informatics*, Kalpa Publications in Computing, 22, EasyChair, 770–780. EasyChair publication page.