Developments in the Built Environment 28 (2026) 101064

Contents lists available at ScienceDirect

Developments in the Built Environment

journal homepage: www.sciencedirect.com/journal/developments-in-the-built-environment

Discretization-based reconstruction of watertight building geometry from raw GIS data for urban CFD simulation

Jinwoo Choi [a], Taehoon Hong [a,*], Hyeonggyun Kim [b], Kwangbok Jeong [b]

[a] Department of Architecture and Architectural Engineering, Yonsei University, Seoul, Republic of Korea
[b] Department of Architectural Engineering, Sejong University, Seoul, Republic of Korea

ARTICLE INFO

Keywords:
Urban CFD
Geometry preprocessing
Watertight geometry
GIS data
Discretization-based reconstruction
Wind environment

ABSTRACT

Urban wind simulations require CFD-ready building geometries, yet real-world GIS datasets often contain geometric inconsistencies that hinder automated preprocessing and mesh generation. This study presents an automated geometry generation pipeline that converts heterogeneous GIS building data into watertight LOD1 models suitable for urban CFD applications. The workflow combines topology-neutral surface sampling, footprint reconstruction, geometry simplification, and automated regularization within a single preprocessing framework. A district-scale case study of approximately 15,000 buildings in Seoul was used to evaluate geometry quality, fidelity, and computational efficiency. The generated models achieved 100% watertightness while preserving building-scale geometric characteristics with median volume fidelity and footprint IoU values of 0.994 and 0.992, respectively. Compared with manual preprocessing workflows, the proposed approach reduced processing time from several days to approximately 190 s. The framework provides a practical foundation for scalable generation of CFD-ready urban geometries, supporting future large-scale airflow simulations and data-driven modeling.

NOMENCLATURE

ABL	Atmospheric boundary layer
AIJ	Architectural Institute of Japan
AWS	Automated weather station
BFS	Breadth-first search
CAD	Computer-aided design
CFD	Computational fluid dynamics
CityGML	City Geography Markup Language
COST	European Cooperation in Science and Technology
CV	Computer vision
DBSCAN	Density-based spatial clustering of applications with noise
DP	Douglas–Peucker
FVM	Finite volume method
GIS	Geographic information system
IoU	Intersection over Union
KMA	Korea Meteorological Administration
KST	Korea Standard Time
LiDAR	Light detection and ranging
LOD	Level of detail
NGII	National Geographic Information Institute
QGIS	Quantum Geographic Information System
RANS	Reynolds-averaged Navier–Stokes
RMSE	Root mean square error
S-DoT	Seoul Data of Things
SIMPLE	Semi-implicit method for pressure-linked equations
STL	Stereolithography
UHI	Urban heat island

(*continued on next column*)

(*continued*)

1. Introduction

1.1. Research background

Rapid urbanization and the proliferation of high-rise, high-density development have intensified urban environmental challenges such as urban heat island (UHI), pollutant dispersion, and degradation of pedestrian-level wind comfort (Brunekreef and Holgate, 2002; Kastner and Dogan, 2023; Choi et al., 2022). Accordingly, the utilization of computational fluid dynamics (CFD) simulation has significantly expanded as an engineering tool to quantitatively predict and evaluate wind flow patterns and microclimate changes in complex urban morphologies (Baitureyeva et al., 2025; Blocken et al., 2013, 2016). CFD can

* Corresponding author.Yonsei University, 50 Yonsei-ro, Seodaemun-gu, Seoul, 03722, Republic of Korea.
E-mail addresses: jinwoo818@yonsei.ac.kr (J. Choi), hong7@yonsei.ac.kr (T. Hong), hgkim@sju.ac.kr (H. Kim), kbjeong7@sejong.ac.kr (K. Jeong).

https://doi.org/10.1016/j.dibe.2026.101064
Received 18 June 2026; Received in revised form 21 September 2026; Accepted 3 October 2026
Available online 5 October 2026
2666-1659/© 2026 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

numerically simulate flow phenomena over large-scale domains where direct field experimentation is practically infeasible, with the advantage of resolving physical quantities such as velocity, pressure, and turbulence in spatial detail. Moreover, CFD is more cost- and time-efficient than wind tunnel experiments for equivalent domains and facilitates repetitive analyses under diverse geometric and meteorological conditions, establishing it as a standard tool for urban wind environment assessment (Blocken et al., 2015; He et al., 2021; Linda et al., 2022). However, to obtain such highly reliable urban CFD analysis results, the construction of a high-quality 3D computational domain that accurately reflects the actual urban geometry must necessarily precede (Franke et al., 2011; Tominaga et al., 2008; Abu-Zidan et al., 2021). Typically, this involves a series of pre-processing tasks, including collecting various types of raw geographic information system (GIS) data, repairing geometric defects, and converting them into the watertight and manifold geometry required by CFD solvers. The problem is that real-world GIS data inherently contains several topological defects, such as gaps, overlaps, and non-manifold edges, which occur during data collection or format conversion processes (Van Den Brink et al., 2013; Ledoux and Meijers, 2011; Liu et al., 2017; Sanyal, 2023). Geometry containing such topological defects causes numerical errors such as negative volumes and inverted elements during the mesh generation stage, making the flow analysis physically impossible. Consequently, this pre-processing workflow has largely relied on manual and iterative modifications by skilled engineers, and it is common for the construction of even a single urban block-scale model to require hours to days of corrective work (Ali et al., 2019). As a result, this manual-based pre--processing bottleneck not only fundamentally limits the application scope of large-scale urban CFD analysis but also acts as a major barrier completely blocking access by non-experts to the field.

Meanwhile, major commercial CFD software packages (e.g., ANSYS Fluent, Siemens Star-CCM+) provide a certain level of geometry correction functionality at the mesh generation stage (e.g., fault-tolerant meshing, shrink-wrap); however, these are fundamentally insufficient for systematically resolving the complex topological defects inherent in urban-scale GIS data. This stems from the structural characteristic that commercial CFD tools must support a wide range of physical phenomena—such as turbomachinery, external aerodynamics, and urban airflow—whose physical scales and geometric features differ substantially. Due to this demand for universality, commercial solvers adopt a design philosophy that delegates the judgment of which geometry details to preserve or simplify for each analysis target to the engineer's empirical-based decision-making, rather than automating it at the software level (ANSYS, 2020; ANSYS, 2017). As a result, this structure poses limitations in different ways for both experts and non-experts in urban CFD analysis. First, for skilled engineers, the diagnosis and repair of geometric defects must be performed iteratively on a building-by-building basis, which requires high time costs and effort for the pre-processing of urban-scale computational domains. On the other hand, for non-experts lacking the empirical know-how of geometry simplification tailored to urban wind environment analysis (e.g., which building details do not affect main flow characteristics and can be removed), this pre-processing step itself acts as a barrier to entry for conducting analyses. In turn, there is a need to develop a robust pre-processing methodology that can automatically generate a 3D urban model optimized for CFD analysis from raw GIS data of various formats and quality levels, without expert intervention.

1.2. Related works

Against this background, attempts to automatically construct or reconstruct 3D geometries of urban environments have been made in various research fields. First, in the field of urban CFD, sensitivity analyses investigating the influence of building geometry's level of detail (LOD) on flow simulation results have constituted the primary research stream. For example, Pađen et al. (2022) systematically compared the effects of LOD 1.2 and LOD 2.2 building representations on pedestrian-level and urban canopy-level flow fields through 1,440 simulations. Also, Karkoulias et al. (2020) quantitatively analyzed the impact of balconies and overhangs on the building envelope on the pedestrian-level wind environment. These studies provided empirical guidelines on the extent to which building geometry can be simplified while maintaining prediction accuracy; however, the pre-processing process for constructing those simplified models still relied on manual work by experts or manual operation of commercial software. Furthermore, although geometry pre-processing is an essential task in all urban CFD studies, its specific procedures or techniques are omitted from descriptions in most studies, perpetuating a structure that relies on individual researchers' tacit experience without accumulating systematic methodologies or technical know-how for pre-processing. Thus, researchers without sufficient technical knowledge or proficiency end up performing analyses on low-quality computational domains where geometric defects are not sufficiently resolved, which can negatively affect the spatial resolution and prediction accuracy of the analysis results (Blocken, 2015).

Second, the GIS and geoinformatics fields have focused on developing tools and workflows that combine 2D GIS data with point clouds to generate 3D urban models. The 3dfier tool developed by Ledoux et al. (2021) automatically generates 3D city models including LOD1 buildings and terrain by extruding 2D topographic polygons based on point cloud elevations, and Peters et al. (2022) extended this approach to automatically reconstruct LOD1/LOD2 models for approximately 10 million buildings across the Netherlands. Meanwhile, Deininger et al. (2020) proposed a semi-automated workflow that applies geometric optimization to CityGML-based 3D city models for use as CFD simulation inputs, and Pađen et al., 2022, 2024 further developed City4CFD, which automatically reconstructs buildings at various detail levels from LOD1.2 to LOD2.2 and integrates them with terrain. City4CFD and 3dfier have demonstrated that automated reconstruction of LOD1/LOD2 city models is feasible. The reconstructed geometries of City4CFD have been directly coupled with CFD mesh generation without additional geometry modification (Pađen et al., 2022, 2024). These tools, however, assume relatively clean input geometry (i.e., well-formed building polygons and point clouds) and rely on the topological consistency of that input during reconstruction. Their extension to heterogeneous, defective GIS datasets—where mesh connectivity itself is compromised—has not been systematically addressed. Furthermore, quality-oriented cleanup of input data — such as the removal of objects that are irrelevant or impractical to simulate — is in practice still performed manually in existing CFD preparation workflows, rather than being integrated into the automated reconstruction pipelines (Pađen et al., 2024).

Third, in the fields of computer vision (CV) and remote sensing, research on automatically reconstructing large-scale urban geometries (3D reconstruction) from point clouds or multi-view images has been actively conducted. For example, Musialski et al. (2013) comprehensively reviewed various urban reconstruction techniques such as structure-from-motion and multi-view stereo, and methodologies for automatically extracting building geometries from airborne LiDAR point clouds have also been widely studied. Recently, 3D reconstruction studies on an urban scale utilizing neural implicit representations have also emerged (Zhang et al., 2024, 2025; Shi et al., 2024). However, the main goal of related research is visual fidelity, meaning the reconstruction of geometries that appear similar to reality based on human vision. Therefore, even if the reconstructed geometry looks visually complete, many critical topological defects such as non-manifold edges, open surfaces, and self-intersections remain in numerical terms (Park et al., 2020). In other words, such geometry reconstruction is suitable for rendering or visualization but fails to guarantee the numerical robustness (i.e., watertightness and manifoldness) required by CFD solvers.

Synthesizing these existing research streams, the following research gaps exist in the field of automated geometry pre-processing for urban CFD. First, an end-to-end automated pipeline capable of consistently processing from raw GIS data input to solver-ready geometry output

without engineer intervention has not been systematically proposed. Existing GIS/geoinformatics tools have automated 3D model generation but lack CFD-specific quality assurance, while pre-processing in existing CFD research remains at the level of manual work through commercial CAD software or ad-hoc scripts limited to specific data formats, leaving a lack of systematic methodologies for handling urban GIS data with varied formats and quality levels. Second, because existing mesh repair or Boolean operation-based approaches fundamentally rely on the topological features of the input data, processing frequently fails under the various defect conditions inherent in real-world GIS data. This suggests the need for a geometry reconstruction strategy that is less dependent on the topological consistency of the input data. Third, in CFD analysis, not only the geometric fidelity of the building shape but also mesh efficiency—which minimizes computational costs by suppressing unnecessary face counts—must be simultaneously considered. Actual urban buildings feature many complex geometries, such as L-shaped, U-shaped, and courtyard forms, and accurately reflecting these irregular shapes while generating lightweight geometry is a task that has not been adequately addressed in previous studies.

1.3. Research aim and contributions

To address these limitations, this study proposes an automated geometry preprocessing pipeline for urban CFD simulations based on computational geometry. The objective of this study is to establish an automated workflow that consistently produces CFD-ready watertight urban geometries from heterogeneous GIS datasets rather than to introduce a new geometry repair algorithm. Accordingly, the contributions of this study are threefold: (i) an automated workflow for generating CFD-ready watertight urban geometries from heterogeneous GIS inputs, including defective pre-existing meshes; (ii) the integration of connectivity-discarding geometry reconstruction and CFD-oriented filtering within a single preprocessing pipeline; and (iii) a demonstration of district-scale geometry generation suitable for downstream CFD mesh generation and large-scale simulation studies. To clarify the positioning of the proposed pipeline, Table 1 compares it against two representative open-source workflows for automated urban 3D geometry generation (i.e., City4CFD and 3dfier). The comparison covers input assumptions, topological dependency, watertightness handling, simplification strategy, physics-aware filtering, and demonstrated application (refer to Table 1). The detailed methodology of the proposed pipeline is described in the following sections, and the generated model is applied as geometric input for CFD simulation targeting an actual urban case study to validate modeling efficiency and analysis suitability. In addition, the developed pipeline is further evaluated through a comparative experiment using the openly published datasets of City4CFD, an open-source workflow developed for urban flow simulations, to verify its practical applicability. The resulting framework is intended to reduce manual preprocessing effort and provide a reproducible pathway for generating CFD-ready urban geometries from heterogeneous GIS datasets, supporting subsequent urban wind environment analyses and applications.

Table 1
Qualitative comparison of the proposed pipeline with representative open-source workflows for automated urban 3D geometry generation.

Aspect	City4CFD (Paden et al., 2022, 2024)	3dfier (Ledoux et al., 2021; Peters et al., 2022)	Proposed pipeline
Input assumptions	Valid, non-overlapping 2D building footprints (ISO, 19107) with height attributes, or footprints combined with classified, properly aligned point clouds	2D topographic polygons combined with a classified point cloud	Heterogeneous GIS inputs, including defective pre-existing meshes with compromised connectivity
Dependence on input connectivity	Reconstruction defined on input footprint polygons; output quality depends on input validity	Extrusion-based; relies on the consistency of input polygons	Input connectivity discarded after component identification; reconstruction from sampled point distributions
Watertightness handling	Watertight outputs reported; validity inspection and automatic validity repair in the extended version	Watertight LoD1 solids by polygon extrusion	Component-level watertightness verification with success-rate accounting
Simplification	RDP-based footprint simplification; graph-cut 3D simplification; isotropic remeshing	Not integrated	Footprint simplification with micro-face suppression
Physics-aware object filtering	Not reported	Not reported	Rule-based filtering by pedestrian-wind relevance thresholds (e.g., minimum building height and footprint area)
Demonstrated application	Campus-scale urban CFD mesh generation (snappyHexMesh)	National-scale 3D city modeling	District-scale urban CFD geometry preprocessing

2. Method and materials

2.1. Overview of the automated modeling pipeline

This study proposes a discretization-based reconstruction pipeline that receives heterogeneous GIS data containing numerous geometric defects as input and automatically generates preprocessed watertight geometry optimized for urban-level CFD flow prediction. Unlike conventional mesh repair approaches that depend on the topological features of input data, the proposed method discretizes geometry into point clouds, thereby discarding the input topology in which such defects reside, and then reconstructs the geometry. Furthermore, physics-aware filtering—domain-specific judgments such as the exclusion of small-scale structures and determination of geometry simplification levels, which were previously performed as separate tasks from geometry repair in existing manual workflows—is internalized as rule-based parameters within the pipeline, enabling geometry reconstruction and analysis-condition filtering to be performed integrally within a single process. This enables the automatic generation of analysis-ready computational domains without relying on engineers' empirical know-how.

The final output of the proposed pipeline corresponds to the LOD1 model as defined in the City Geography Markup Language (CityGML) standard (Kolbe et al., 2005; Biljecki et al., 2016a). LOD1 represents buildings as prismatic blocks with flat roofs by vertically extruding building footprints, excluding roof geometry and detailed building envelope elements (e.g., balconies, windows, protrusions) (Ledoux and Meijers, 2011; Chajaei and Bagheri, 2025; Biljecki et al., 2014). LOD1 models are widely adopted in urban-scale CFD and environmental simulations including urban wind environment, pollutant dispersion, and solar irradiance analyses, based on existing research findings that the macroscopic characteristics of the flow field in pedestrian-level wind environment analysis are dominated by the overall volume (blockage) and arrangement of buildings, while the influence of building envelope details on flow structures at pedestrian height is relatively limited (Van Den Brink et al., 2013; Biljecki et al., 2016b; Becker, 2009). Accordingly, this study designed the pipeline under the premise that LOD1-level geometry representation constitutes the optimal simplification level for urban wind environment analysis. To this end, the proposed automated pipeline comprises the following four steps: (i) individual building

identification; (ii) surface sampling and normalization; (iii) footprint extraction using concave hull algorithm; and (iv) geometry simplification, extrusion, and regularization (refer to Fig. 1).

2.2. Individual building identification

Urban-scale GIS data is provided in a format where information on numerous buildings is included in a single file, so separating this into individual building units is an essential prerequisite for geometry reconstruction. However, GIS data takes various formats depending on the collection and distribution methods, and the types and presence of information defining boundaries between buildings differ according to the input data format. Therefore, the pipeline is designed to automatically determine the geometry input format and select an appropriate building identification path before performing pre-processing tasks.

First, if the input geometry is mesh-based data such as CityGML, Shapefile extrusion (.shp), Stereolithography (.stl) or object (.obj), this mesh inherently contains the connectivity of faces and edges. Based on this connectivity information, the mesh can be represented as a dual graph $G = (F, E)$ (AbuBaker et al., 2007). Here, F is the set of faces, and E is the adjacency relation between faces, which can be defined as adjacent when two faces f_i and f_j share one or more edges (refer to Eq. (1)). Also, taking the transitive closure of this adjacency relation, a connected component C_k can be defined as a set of mutually reachable faces through a chain of shared edges (Dillencourt et al., 1992). Specifically, this means that for any two faces f_i and f_j belonging to C_k, there exists a finite adjacency path from f_i to f_j (refer to Eq. (2)). The search for these connected components is performed using a breadth-first search (BFS)-based flood-fill traversal (Fishkin et al., 1985). In detail, this algorithm selects one unvisited face as a seed, adds adjacent faces sharing edges with that face to a queue, and repeats this process until the queue is exhausted to complete one connected component. Then, a new seed is selected from the remaining unvisited faces to explore the next component, and this is repeated until all faces are visited. Since each face and edge is processed exactly once, the time complexity of the entire process is $O(|F|+|E|)$, linearly proportional to the number of faces and edges, ensuring high computational efficiency even in large-scale urban data. Physically separated buildings do not share edges, so they can be automatically identified as individual components. As the majority of urban GIS data is provided in mesh-based formats, this identification process corresponds to the primary path most frequently applied in practice.

$$A\left(f_i, f_j\right) = \left\{1 \text{ if } \exists e \in E_{mesh} \text{ s.t. } e \in \partial f_i \cap \partial f_j; 0 \text{ otherwise}\right\} \tag{1}$$

$$\forall f_i, f_j \in C_k, \exists \left(f_i = g_0, g_1, \cdots, g_n = f_j\right) \text{ s.t. } A(g_t, g_{\{t+1\}}) = 1, \forall t \in \{0, \cdots, n-1\} \tag{2}$$

Where $A(f_i, f_j)$ stands for the adjacency function between faces f_i and f_j, E_{mesh} for the set of edges in the mesh, ∂f for the set of edges bounding face f, $g_0, g_1, \ldots, g_n$ stands for a finite sequence of faces forming an adjacency path from f_i to f_j within C_k.

Meanwhile, if the geometry input is point cloud-based (e.g., LiDAR raw scan, photogrammetry), there is no connectivity between faces or edges, so the aforementioned connectivity-based separation cannot be applied. In this case, building identification is based on the spatial density of points, for which this study applied the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm (Schubert et al., 2017; Campello et al., 2013). DBSCAN forms clusters based on data density and is suitable for processing large-scale urban data where the number of buildings cannot be known in advance.

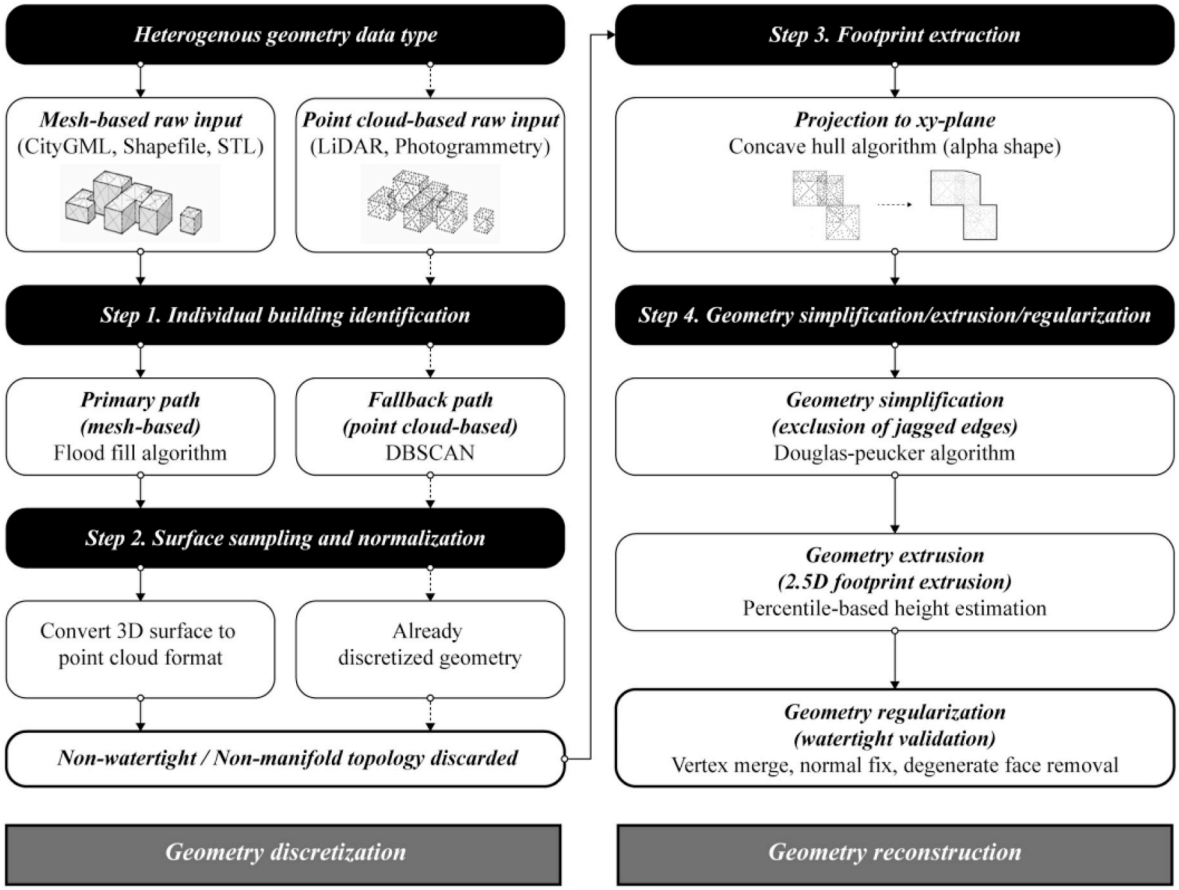


Fig. 1. Framework of the urban geometry reconstruction pipeline.

Additionally, it can automatically remove points that do not meet the density criteria by classifying them as noise, making it effective for filtering sporadic outlier points included in LiDAR data (e.g., trees, vehicles, ground reflections). For an arbitrary point p to be defined as a core point in the DBSCAN algorithm, the condition is that at least *MinPts* points exist within the ε-neighborhood of radius ε (refer to Eqs. (3) and (4)).

$$N_\varepsilon(p) = \{q \in P | dist(p,q) \le \varepsilon\} \tag{3}$$

$$|N_\varepsilon(p)| \ge MinPts \tag{4}$$

Where $N_\varepsilon(p)$ stands for the set of points within the ε-neighborhood of point p, $dist\ (p, q)$ for the Euclidean distance between points p and q, ε for the maximum radius of the neighborhood, and *MinPts* for the minimum number of points required to form a dense region.

2.3. Surface sampling and normalization

Typical GIS data contains numerous numerically unstable topological defects such as gaps, overlaps, and non-manifold edges due to geometric errors arising from 3D modeling or format conversion processes, and these inherent defects constitute the primary factor causing conventional Boolean operations and automated healing algorithms to fail. Such correction failures occur because conventional approaches rely on the topological connectivity of the input geometry to locally correct defects, and when the connectivity itself is compromised, no reference point for correction can be established. To address the fundamental limitations of topology-dependent approaches, this study performed a normalization process that discretizes the surface into a point cloud for each mesh component identified as an individual building. This process corresponds to geometry discretization, converting a continuous vector-based surface into a finite set of points. The process of defining the surface S of a geometry as a discretized point set P is mathematically expressed as follows (refer to Eqs. (5) and (6)).

$$P = \{p_i \in \mathbb{R}^3 | p_i \ lies\ on\ S, i = 1, 2, \ldots, N\} \tag{5}$$

$$N \approx A_{Building} \times \rho \tag{6}$$

Where P stands for the set of sampled points, p_i for the i-th point with coordinates (x, y, z), $\mathbb{R}$ for the set of real numbers, S for the surface of the individual building component, N for the total number of sampled points, $A_{Building}$ for the surface area of the building, and ρ for the sampling density (points/m^2).

Through this process, geometric information of the shape is preserved, while the complete topological representation of the input mesh—including any gaps, overlaps, and non-manifold edges it contains—is intentionally discarded via surface sampling and is therefore not carried over to the subsequent stages. For point cloud-based inputs such as LiDAR scan data, the input data is already in point cloud format, so the clustered point set is directly used as input for subsequent stages without separate surface sampling. All subsequent reconstruction processes are performed based solely on the spatial distribution of points rather than existing topological information, thereby producing consistent results across the heterogeneous GIS and point-cloud inputs investigated in this study.

2.4. Footprint extraction using concave hull algorithm

The process of extracting 2D footprints, which form the basis of LOD1 models, from the constructed individual building clusters was performed. First, the set of points in each cluster was projected onto the xy-plane to generate a 2D point set, from which a hull algorithm was applied to extract the building's footprint boundary.

The convex hull algorithm widely used in previous studies for footprint boundary extraction is a method that generates the smallest convex polygon containing the point cloud (Barber et al., 1996). While this approach is computationally efficient, it generates overly simplified footprints for buildings with re-entrant corners such as L-shaped, U-shaped, and courtyard buildings by encompassing concave regions. Such over-simplification results in the loss of key geometric features that directly influence wind flow patterns within street canyons, constituting an unacceptable geometric distortion in the context of urban wind environment analysis.

Therefore, to overcome this, an alpha shape-based concave hull algorithm was applied in this study (Asaeedi et al., 2017). The alpha shape refers to the boundary line formed when an open ball $B(b, \alpha)$ with radius α moves along the boundary surface without penetrating the interior of the point set (refer to Eq. (7)). Here, α is the key hyperparameter controlling the complexity of the extracted boundary; as $\alpha \to \infty$, the result converges to a convex hull, and concave features are lost; as $\alpha \to 0$, the boundary penetrates into fine gaps between individual points, generating a discontinuous shape. Meanwhile, for small buildings where the projected 2D point set contains fewer than 6 points even after surface sampling, convex hull was applied instead of concave hull, as the mathematical definition of alpha shapes can lead to numerically unstable boundary extraction or degenerate polygon generation when the number of points is extremely small.

$$\partial S_\alpha = \left\{ x \in \mathbb{R}^2 \middle| \exists (b, \alpha) s.t. x \in \partial B(b, \alpha)\ and\ B(b, \alpha) \cap P = \varnothing \right\} \tag{7}$$

Where ∂S_α stands for the boundary of the alpha shape, $B(b,\alpha)$ for the open ball with center b and radius α, and P for the set of projected 2D points.

2.5. Geometry simplification, extrusion, and regularization

The extracted raw concave hull boundaries may contain unnecessarily complex jagged edges due to the influence of sampling noise. These fine irregularities cause extremely small grid elements (micro-elements) during the CFD mesh generation process, significantly degrading computational efficiency (Piepereit et al., 2018). In particular, if micro-faces with areas significantly smaller than the grid resolution remain on the building surface, an imbalance in the size ratio with adjacent grids and an increase in skewness occur during the mesh generation process. To overcome this, this study applied the Douglas-Peucker (DP) algorithm to the extracted footprint for each building to remove unnecessary vertices within a tolerance range while maintaining the characteristic topology of the building geometry (Douglas and Peucker, 1973; Shi and Cheung, 2006). In detail, the DP algorithm calculates the perpendicular distance d_i of each intermediate point from the line segment connecting the start and end points, and recursively removes points whose distance falls below the tolerance ε_{dp} (refer to Eq. (8)). This process fundamentally suppresses the occurrence of micro-faces by straightening jagged edges, thereby securing the stability of downstream mesh generation.

$$d_i = \frac{|(y_2 - y_1)x_i - (x_2 - x_1)y_i + x_2 y_1 - y_2 x_1|}{\sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}} < \varepsilon_{dp} \tag{8}$$

Where d_i stands for the perpendicular distance of point p_i from the line segment defined by p_1 and p_2, and ε_{dp} for the user-defined tolerance.

Next, footprint extrusion was performed to convert the simplified 2D footprint into an LOD1 3D solid model. The height H of each building cluster was calculated utilizing the percentiles for the z-coordinates of the points within the cluster (refer to Eq. (9)). The use of the 99th upper percentile and 5th lower percentile is to stably estimate the actual height of the building while suppressing the influence of extreme outlier points (e.g., outliers caused by rooftop structures, ground reflection errors) that may be included in the LiDAR data. For mesh-based inputs, the sampled surface points include the facades, and the percentile-based height definition of Eq. (9) is applied directly. For point-cloud inputs derived

from classified airborne LiDAR, where wall returns are largely absent and the building points are dominated by roof returns, the lower percentile of a cluster corresponds to the eave level rather than the ground. In this case, the base elevation of each cluster is instead estimated as the median elevation of the accompanying ground points in the immediate vicinity of the cluster (i.e., implemented as the cluster bounding box expanded by the ground buffer of 5 m), and the pipeline accepts the ground point cloud as an optional input for this regime.

$$H = Z_{99\%} - Z_{5\%} \tag{9}$$

Where H stands for the height of the building, and $Z_{99\%}$ and $Z_{5\%}$ for the 99th and 5th percentiles of the z-coordinates within the cluster, respectively.

Finally, automated regularization was performed on the 3D geometry generated through extrusion. This process utilized the Python Trimesh library to (i) merge duplicate vertices, (ii) consistently unify the directions of face normals, (iii) remove degenerate faces, and (iv) automatically verify that the watertight condition is maintained after removal, thereby securing a watertight closed manifold. A watertight condition allows the generated model to be transferred to the CFD solver's mesh generation stage through the standard workflow without relying on mesher-side repair, and the pipeline includes a quality control procedure that automatically verifies watertightness for each building and excludes buildings that do not satisfy the condition from the final output. Accordingly, Fig. 2 visualizes the individual stages of this procedure for a U-shaped building: the sampled point set, its xy projection, the alpha-shape footprint, the Douglas-Peucker simplification, the regularized footprint, and the extruded watertight LOD1 block. The concave-hull ratio α and the simplification tolerance ε govern the balance between geometric fidelity and footprint compactness. Overly small α values follow the sampling noise of the point distribution, whereas large values approach the convex hull and lose concave footprint features; small ε values retain boundary detail, whereas large values produce compact footprints at the cost of cutting concave corners. The default configuration ($\alpha = 0.2$, $\varepsilon = 0.5$ m) preserves the concave outline of the footprint while suppressing sampling-induced boundary noise.

2.6. Physics-aware hyperparameter setting

Geometry simplification and object filtering applied in this pipeline are justified by domain-specific rules based on the physical characteristics of urban wind environment analysis. These rules correspond to internalizing decision-making (e.g., which structures to include, to what extent to simplify specific geometric elements) that relied on engineers' empirical judgments in existing manual workflows into quantitative parameters. Through this, the pipeline performs geometry reconstruction and physics-aware filtering within a single process without separate manual work. This section comprehensively describes the parameter setting values and physical bases of each of these processes.

First, regarding the appropriateness of LOD1 expression, this pipeline represents buildings as prismatic blocks with flat roofs. In pedestrian-level wind environment analysis, the primary region of interest is the flow field at 1.5–2.0 m above ground, and the flow characteristics of this region are dominated by the blockage effect of the overall volume and arrangement of buildings (Blocken et al., 2015, 2016). According to existing research, the influence of detailed roof geometry differences on pedestrian-level wind speeds is reported to be minor relative to building height, and LOD1 models have accordingly been adopted as standard in numerous urban CFD studies (Chajaei and Bagheri, 2025; Biljecki et al., 2016b). However, for cases where roof geometry directly influences analysis results, such as flow field analysis around building rooftops or wind pressure distribution analysis, LOD2 or higher representation may be required, which is separately discussed in the Discussions section as a limitation on the pipeline's scope of application.

Second, the surface sampling density $\rho = 10$ pt/m^2 was set to ensure sufficient point density for concave hull boundary extraction. For building footprint boundaries to be faithfully extracted, a sufficient number of points must be distributed near the boundary, and 10 pt/m^2 stably satisfies this condition by generating 100–2,000 points per building relative to the surface area of typical urban buildings (Haala and Kada, 2010).

Third, regarding DBSCAN parameters, $\varepsilon = 2.5$ m is based on the fact that the minimum separation distance between adjacent buildings in urban environments is typically regulated to be more than 2 m by building codes and right-to-light standards (MOLIT, 2018). By setting ε below this separation distance, errors where physically separated buildings are merged into the same cluster can be prevented. Also, $MinPts = 30$ was set considering the typical point density of LiDAR sensors (4–20 points/m^2), at a level where the minimum target building for expression based on LOD1 (6 m^2 of footprint area) is recognized as a valid cluster (Peters et al., 2022).

Fourth, the α ratio = 0.2 for the concave hull was set as a balance point that preserves the main concave features of buildings while preventing discontinuities caused by noise. The typical re-entrant depth of

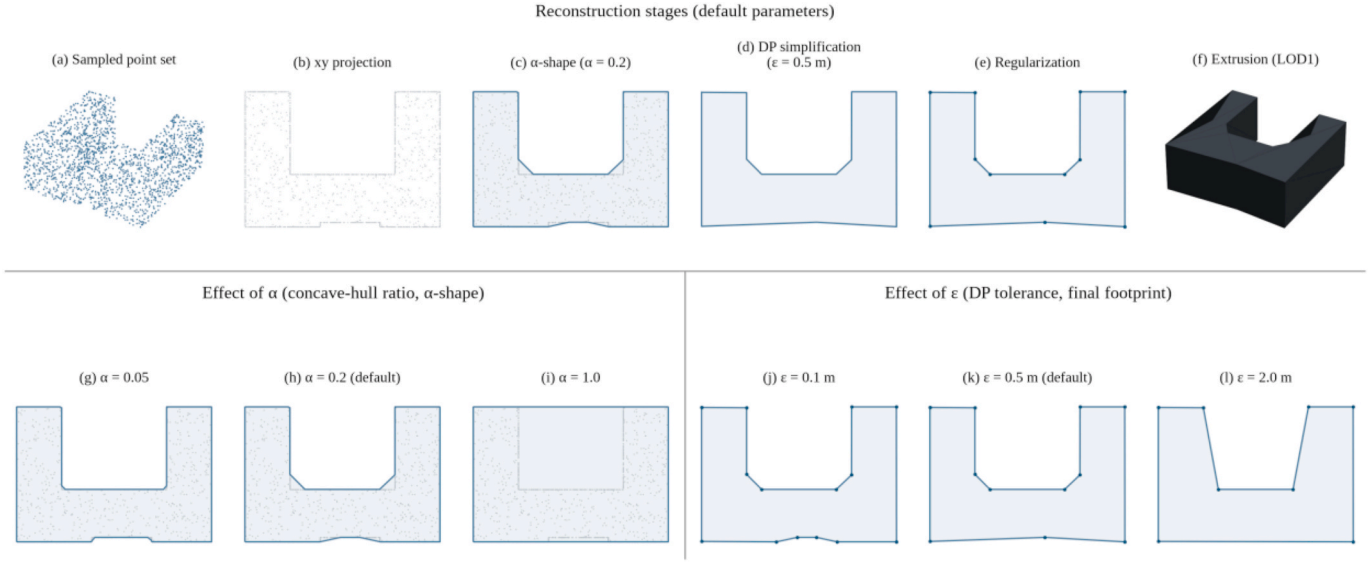


Fig. 2. Reconstruction of a U-shaped building by the proposed pipeline and effects of parameters.

urban buildings (e.g., concave depth of L-shaped buildings) is distributed in the 3–10 m range, and geometric features in this range are directly involved in flow patterns within street canyons (e.g., vortex formation, channeling effect) (Vardoulakis et al., 2003; Zhong et al., 2017; Ding et al., 2019). The set α ratio of 0.2 corresponds to a range where concave features of this scale are faithfully extracted while false concavities due to fine gaps between points do not occur.

Fifth, the DP tolerance $\varepsilon_{dp} = 0.5$ m is justified by the following compound rationale. First, the grid resolution for pedestrian-level wind analysis is typically set at the 1–2 m level according to COST 732 and AIJ guidelines, and ε_{dp} of 0.5 m corresponds to a sub-grid scale displacement of 25–50% of this grid resolution, so its influence on analysis results remains within grid discretization error (Franke et al., 2011; Tominaga et al., 2008). Also, the positional accuracy requirement of the CityGML LOD1 standard is 5 m or less, and a geometric displacement of 0.5 m achieves a conservative tolerance at the 10% level (Kolbe et al., 2005). Moreover, the distance threshold for geometry simplification to ensure CFD mesh quality (non-orthogonality, skewness) has been estimated in the range of 0.01–0.1 times the characteristic length, and the ε_{dp} of this study represents the median of the allowable range at 0.05 times the average characteristic length of the target buildings (Park et al., 2020). In turn, structural features of buildings that directly influence the flow field, such as re-entrant corners and setback geometry, typically have scales of several meters or more, so the preservation of these features is guaranteed at the 0.5 m tolerance.

Sixth, the proposed pipeline excludes structures with a height of $H_{\min} <$ 2.0 m and objects with a calculated footprint area of $A_{fp,min} < 8\ m^2$ from geometry reconstruction as physics-aware filtering. The height of interest for pedestrian-level wind environment analysis is 1.5–2.0 m above ground, and the blockage effect on the flow field of structures at or below this height is negligible compared to adjacent buildings (Tominaga et al., 2008; ANSYS, 2017). Also, GIS data includes fences, retaining walls, street facilities, and garden structures as independent polygons in addition to buildings, and many of these are non-building objects with a height of 2 m or more that cannot be excluded by height-based filtering alone. Recent urban CFD preprocessing studies, including City4CFD, have also reported that excluding very small-scale polygons through footprint simplification contributes to improving the stability of the meshing process (Pađen et al., 2022). The 8 m^2 applied in this study is a conservative value compared to the CityGML standard, set to effectively exclude non-building objects while preserving independent structures of the minimum scale that can be considered actual buildings (e.g., small guardhouses, electrical rooms). However, the set thresholds are designed as user-defined parameters adjustable according to analysis purposes, and can be adjusted for analyses where the influence of small-scale structures is important (e.g., micro flow analysis around road accessories). The rationale for the above parameter settings is summarized in Table 2. The entire pipeline is implemented in Python 3. Mesh loading, surface sampling, connected-component decomposition, and STL export are performed using the Trimesh library; density-based clustering using the DBSCAN implementation of scikit-learn; concave hull extraction and Douglas–Peucker footprint simplification using Shapely; and height statistics using NumPy. The complete source code, the exact configuration used in this study, the baseline, reference, and modeled geometries of the case study area, and scripts for reproducing the geometric quality metrics reported in Table 4 are publicly available in a versioned repository at https://github.com/jinwoo818/Discretization-based-reconstruction-of-raw-GIS-to-watertight-geometry.

Table 2
Summary of the hyperparameters and their physical justification.

Step	Parameter	Value	Physical justification	Reference
Sampling	ρ	10 pt/m^2	Sufficient density for concave hull boundary extraction	Haala and Kada (2010)
Clustering	ε	2.5m	Minimum building separation distance according to building code and setback regulation	MOLIT (2018)
	MinPts	30	Minimum detectable building footprint at typical LiDAR density (≈6 m^2, LOD1 threshold)	Kolbe et al. (2005)
Footprint extraction	α ratio	0.2	Preservation of concave features (re-entrant depth 3–10 m) without boundary discontinuity	Edelsbrunner et al. (1983)
Simplification	ε_{dp}	0.5 m	Sub-grid scale relative to pedestrian-level CFD resolution (1–2 m); 0.05×characteristic length	(Franke et al., 2011; Tominaga et al., 2008; Park et al., 2020; Piepereit et al., 2018)
Height estimation	Percentile range	$Z_{5\%}$-$Z_{99\%}$	Outlier suppression while retaining effective building height	(Dorninger and Pfeifer, 2008; Rottensteiner, 2003; Biljecki et al., 2017)
Filtering	$H_{\min}$	2.0 m	Below pedestrian-level interest height (2.0 m); excludes non-building structures (walls, fences)	(Blocken, 2015; Ali Smail et al., 2024)
	$A_{fp,min}$	8 m^2	Conservative to CityGML LOD1 minimum	(Kolbe et al., 2005; Biljecki et al., 2015; Beil and Kolbe, 2017)

2.7. Geometric quality and fidelity metrics

To quantitatively evaluate the output quality of the proposed pipeline and systematically compare it with the original data and manual cleanup results, two categories of metrics—geometric quality metrics and geometric fidelity metrics—were defined. Geometric quality metrics evaluate the extent to which topological defects inherent in input data are resolved through each processing stage; they are applied identically to three datasets—(i) raw GIS geometry (baseline data); (ii) manually cleaned geometry (reference data); and (iii) pipeline output (modeled data)—to compare the absolute numerical stability of each geometry. Geometric fidelity metrics evaluate how faithfully the pipeline's geometry reconstruction preserves the geometric characteristics of the original buildings, computed based on building-by-building correspondence between input and output buildings. The individual indicators in each category are defined below.

First, the following four indicators were defined as geometric quality metrics. All ratio metrics are computed over the connected components of the evaluated geometry of each dataset—that is, the geometry as exported for comparison—after merging coincident vertices, with the components decomposed under manifold-edge adjacency and residual fragments with fewer than eight faces included in the count. Since the pipeline may exclude input objects through its physics-aware filtering and quality-control procedures, the number of objects excluded at each stage is reported separately in the case study, so that the quality of the exported geometry and the reconstruction yield of the workflow can be evaluated independently.

- *Watertightness rate*: *Watertightness rate* is defined as the proportion of buildings satisfying the watertight condition out of the total number of building objects. For a mesh to be watertight means that all edges constituting the mesh are shared by exactly two faces, completely separating the interior and exterior topologically. Verification of watertightness is performed by checking the satisfaction of the Euler

characteristic (i.e., χ = vertices − edges + faces = 2), which is a necessary condition for a closed orientable manifold, along with the consistency of face normals. Watertightness is a prerequisite for the standard watertight-geometry meshing workflows of major CFD solvers. Although commercial solvers provide geometry-correction functionality at the mesh generation stage, such mesher-side repair handles only limited defects and its geometric outcomes are not controlled at the preprocessing stage. Guaranteeing watertightness at the geometry-preprocessing stage therefore ensures that volume mesh generation can proceed through the standard workflow in a predictable and tool-independent manner.

- *Non-manifold edge ratio*: Non-manifold edge ratio is defined as the proportion of edges shared by three or more faces relative to the unique edge count of all meshes. In a normal closed surface (two-manifold), all edges must be shared by exactly two faces, but non-manifold edges that do not comply with this can frequently occur due to overlapping faces in GIS data or incomplete manual cleanup processes. These non-manifold edges make it impossible to determine the orientation of the surface during mesh generation, acting as a major cause of numerical errors such as negative volumes.
- *Degenerate face count*: *Degenerate face* count is reported as the number of triangular faces whose three vertices lie on an identical point or line, resulting in an actual area of zero. Such degenerate faces cause the Jacobian determinant for mapping the grid from physical space to computational space to be 0, making the inverse matrix unobtainable, which induces numerical errors in the finite volume method (FVM)-based CFD solver and makes analysis impossible.
- *Micro-face ratio*: Micro-face ratio is computed as both the ratio of buildings containing faces with areas significantly smaller than the grid resolution and the ratio of such micro-faces to the total number of faces. In this study, considering the typical grid resolution of pedestrian-level CFD, 0.25 m^2 (i.e., 0.5 m × 0.5 m), corresponding to the sub-grid scale of the surface cell, was set as the micro-face threshold. Unlike degenerate faces, these micro-faces are mathematically valid; however, during CFD mesh generation, they force the creation of extremely small surface cells, causing size ratio imbalance with adjacent grids, increased skewness, and time step constraints.

Meanwhile, to evaluate how faithfully the pipeline's geometry reconstruction preserves the geometric characteristics of the original buildings, three geometric fidelity metrics were additionally defined. These metrics perform building-level comparison of modeled data against reference data, with inter-building correspondence established based on nearest centroid matching on the *xy*-plane.

- *Volume fidelity*: *Volume fidelity* is calculated as the volume ratio of corresponding building pairs ($V_{modeled}/V_{reference}$). The closer this ratio is to 1.0, the more faithfully the pipeline output reproduces the 3D volume of the building, and the main factor for deviation is the volume change caused by the LOD1 conversion.
- *Footprint IoU*: Footprint intersection over union (IoU) is defined as the value obtained by dividing the intersection area by the union area between the 2D footprint polygons of corresponding building pairs. IoU is an indicator that simultaneously reflects the positional accuracy and shape similarity of the geometry, and is particularly effective in evaluating the degree of preservation of concave features (e.g., re-entrant corners, setback areas). An IoU of 1.0 means the two footprints match completely.
- *Height accuracy*: *Height accuracy* is computed as the absolute error between the height estimates of corresponding building pairs ($|H_{modeled} - H_{reference}|$). Since the pipeline's height estimation uses the percentile-based method presented in Eq. (9), this metric serves to verify whether systematic deviations exist due to differences from the original data's height representation method.

3. Case study description

This study selected the Mapo-gu (district) in Seoul, South Korea as the target area for the practical applicability and validation of the proposed pipeline. Mapo-gu is a typical high-density mixed-use urban area where high-rise mixed-use complexes, mid-rise commercial facilities, and low-rise residential buildings coexist, making it suitable for verifying the pipeline's versatility due to its inclusion of diverse building forms and densities. The source data used for validation was the GIS building integrated data provided by the National Geographic Information Institute (NGII) (Korea Land and Housing Corporation, 2018; MOLIT, 2021). This data was provided in Shapefile (.shp) format as a 2.5D format containing building footprint polygons and height attributes. To define the target area, QGIS was used to set a circular region of 4 km diameter centered on a point of interest and extract building data within that range, with a default single-story building height (3 m) applied for buildings with missing height information. The final extracted target area contained a total of 15,235 GIS polygons (components), and road and terrain data included in the source data were removed beforehand as they are not subject to building geometry reconstruction. From this, each polygon was extruded based on its height attribute and converted to a uniform STL surface geometry to construct the raw GIS-based baseline data (refer to Fig. 3).

Next, as reference data for comparatively evaluating the output quality of the pipeline, manual-based geometry refinement was performed separately on the raw GIS data of the same study area. Manual cleanup was a multi-stage process utilizing Blender and ANSYS SpaceClaim. After manual identification and removal of non-building objects (e.g., vegetation, road accessory facilities, sign structures), resolving overlapping faces, filling open holes, and unifying face normals in Blender, residual topological defects were finally resolved in ANSYS SpaceClaim. This process took several days of work, which is due to the nature of the task requiring iterative diagnosis and correction of topological defects on a building-by-building basis. Meanwhile, the manual cleanup was focused on resolving topological defects, and physics-aware filtering such as removing small-scale structures or adjusting geometry simplification levels according to the analysis purpose was not applied separately. This reflects the general practice in existing manual workflows where geometry repair and geometry adjustment according to analysis conditions are performed as separate tasks.

Finally, the modeled data—the pipeline-based output—was generated using the same raw GIS data as the baseline dataset, following the methodology described in Section 2. Since the input data for this case study, the NGII building integrated data, was identified as mesh format extruded from Shapefiles, the mesh connectivity-based component separation path was applied at the building identification stage, followed by sequential execution of surface sampling, footprint extraction, geometry simplification, extrusion, and regularization for each building. The hyperparameter settings applied during pipeline execution used the values listed in Table 2.

Ultimately, three types of datasets—baseline, reference, and modeled—were constructed for the same study area, and a quantitative comparative analysis between the datasets and an evaluation of the pipeline's performance were performed by applying the geometric quality and fidelity metrics (refer to Table 3). In addition to this, to verify the practicality of the pipeline's output, CFD simulations were performed using reference data and modeled data respectively as input geometries under the same flow conditions, and the equivalence of flow prediction was evaluated through comparison with measured wind speed data from weather stations within the target region.

4. Results

4.1. Comparative analysis of geometry quality

Table 4 presents the results of a comparative analysis based on geometric quality metrics for a total of three datasets as baseline, reference, and modeled data.



Fig. 3. Raw GIS-based geometry of the case study area.

Table 3
Summary of three comparative geometry datasets.

Dataset	Source	Processing method
Raw GIS	NGII building GIS data (Shapefile)	Height extrusion only (3m default for missing value)
Manual cleanup	Same raw GIS data	Blender and SpaceClaim (multi-stage repair)
Pipeline output	Same raw GIS data	Proposed automated pipeline (CFD-oriented geometry generation)

First, examining the geometric quality of the raw GIS data, the extrusion of individual buildings generates mathematically complete prisms, so the majority of buildings satisfied the watertight condition. However, when combined into a single mesh, watertightness decreased to 98.8% due to shared walls between adjacent buildings, and a total of 568 non-manifold edges were generated. This is attributable to the frequent occurrence of extremely short separation distances or shared walls between buildings in urban environments, and such inter-building topological conflicts constitute a primary cause of mesh generation errors. Additionally, degenerate faces were observed in 293 buildings in the raw GIS data, and micro-faces with areas below 0.25 m^2 were detected in 1,905 buildings, representing 12.2% of all buildings. These micro-faces result from the original GIS data's footprint polygons containing excessively fine boundary lines that are directly transferred without modification during the extrusion process, forcing the creation of micro-cells during mesh generation and causing unnecessary computational cost increases and mesh quality degradation. The component count of the baseline data was determined as 15,659—approximately 400 more than the original 15,235 GIS polygons—as some complex geometries containing multipart geometries or interior holes were decomposed into surface units during the conversion to extrusion and STL surface geometry representation.

Table 4
Geometric quality comparison of baseline, reference, and modeled data.

Data type	Baseline (Raw GIS)	Reference (Manual cleanup)	Modeled (Pipeline output)
Components	15,659	15,072	15,056
Total faces	365,668	284,223	273,432
Watertightness	98.8%	96.9%*	100%
Non-manifold edges	568	527*	0
Degenerate face bldgs.	293 (1.9%)	16 (0.1%)	0 (0%)
Micro-face bldgs.	1,905 (12.2%)	745 (4.9%)	188 (1.2%)
Total micro-faces	8,608 (2.4%)	2,920 (1.0%)	394 (0.1%)

Note: All values are computed with the released measurement script after merging coincident vertices; components are decomposed under manifold-edge adjacency, and the watertightness rate and the percentages in parentheses are calculated over all connected components, including residual fragments with fewer than 8 faces.
Note*: The reference geometry retains 527 non-manifold edges, generated at coincident walls between adjacent buildings, and 465 components that do not satisfy the watertightness criterion (i.e., pieces of buildings connected by such shared walls, and residual open shells and fragments left by the manual workflow).

Second, for the reference building data generated through manual cleanup, the first-stage cleanup process using Blender—including manual removal of non-building objects, overlapping face resolution, open hole filling, and face normal unification—resulted in a substantial reduction of degenerate-face buildings from 293 to 16, and the micro-face ratio also improved from 12.2% to 4.9% compared to the baseline. However, during the repair process in Blender, non-manifold edges were incidentally generated in some buildings, and a temporary degradation of watertightness was observed; at this intermediate stage, 611 non-manifold edges and 7,151 open edges remained across the working geometry. These intermediate-stage defects were substantially resolved through the geometry repair function of ANSYS SpaceClaim, which reduced the open edges to 519. Nevertheless, the final exported geometry still contains 527 non-manifold edges—all of which originate from the earlier stage and arise at coincident walls between adjacent buildings—so that the watertightness rate of the reference data is 96.9%. Such inter-building topological conflicts cannot be resolved by repair operations that treat each building independently, as they require modifying the shared boundaries between adjacent buildings themselves. The residual defects did not prevent volume mesh generation in the subsequent CFD analysis, as they were absorbed by the geometry-correction functionality of the commercial mesher at the mesh generation stage; however, this implies that the meshing process relies on mesher-side repair whose geometric outcomes are not controlled at the preprocessing stage. In addition, due to the structural limitations of manual cleanup by a human expert, 16 degenerate-face buildings and 2,920 micro-faces across the entire domain remained uncorrected, and these defects can serve as potential factors for mesh quality degradation and increased solver computational cost during CFD mesh generation.

Meanwhile, the modeled data based on the proposed pipeline showed the most favorable results across all geometric quality metrics. This geometry achieved 100% watertightness along with 0 non-manifold edges and degenerate faces, and the micro-face ratio was also the lowest among the three datasets at 1.2%. These results are attributable to the pipeline adopting a reconstruction approach that does not depend on the topology of the original geometry but reconstructs from point clouds, and to the DP algorithm effectively suppressing micro-face generation through boundary straightening. Furthermore, the pipeline automatically excluded small-scale structures below 2 m in height and non-building objects through physics-aware filtering, resulting in the component count being determined as 15,056, slightly fewer than the manually cleaned data. Specifically, of the 15,251 building candidates identified from the input components, 15,056 buildings (98.7%) were successfully reconstructed; 188 candidates were excluded by the physics-aware filtering (1 below the minimum building height and 187 below the minimum footprint area) and 7 failed reconstruction, and all 15,056 reconstructed buildings passed the watertightness verification. This indicates that analysis condition-specific object filtering, which would need to be performed as a separate task from geometry cleanup, was automatically performed within the pipeline.

In terms of total face count, both reference data (284,223; 22.3% reduction) and modeled data (273,432; 25.2% reduction) achieved reduction compared to the baseline data (365,668), with the modeled data exhibiting a higher face reduction rate. This is because the pipeline simplifies fine irregularities of the original polygon during the footprint reconstruction process, thereby securing mesh efficiency. In terms of processing time, manual cleanup required approximately 7 days, whereas the pipeline processed approximately 15,000 buildings across the same domain in 190 s on a workstation equipped with an Intel Core i7-12700KF CPU (3.61 GHz) and 32 GB of memory. The reported processing time is the total wall-clock time of a single pipeline execution, including input loading and output export, measured with the Python time module. It covers the geometry-preprocessing stage of the pipeline only and does not include the downstream volume meshing, whose wall-clock time is determined by the selected mesher, meshing settings, and hardware. This reduces an effort that requires manual work on the order of several days to a matter of minutes.

Next, geometric fidelity metrics were calculated based on the modeled data against the reference data (refer to Table 5). First, the median value of volume fidelity was 0.994, extremely close to 1.0, indicating that the pipeline output almost accurately reproduces the 3D volume of the buildings. In particular, as 99.8% of all buildings showed a volume ratio within ±10%, it was confirmed that the volume change based on the LOD1 conversion under the flat roof assumption is negligible for most buildings. Next, as the median of footprint IoU was calculated to be 0.992, it showed that the pipeline's footprint reconstruction preserved concave features such as re-entrant corners and setback areas, reproducing the original shape with high accuracy. In addition, the proportion of buildings with footprint IoU > 0.9 across all buildings reached 100.0%, so no substantial geometric distortion was observed according to the selected fidelity metrics. As the median of height accuracy was calculated to be 0 m, it was confirmed that the percentile-based height estimation yielded values substantially identical to the heights of the manually cleaned data.

4.2. CFD validation for practical flow prediction

To verify the practicality of the pipeline output, CFD simulations were performed applying the reference data and modeled data respectively as input geometries, and comparisons with measured wind speed data from weather stations within the target area were conducted. For the measured data, the Smart Seoul City Data Sensor (S-DoT) environmental information provided by the Seoul Open Data Plaza was used (Seoul Metropolitan Government, 2023, 2026). S-DoT is an urban environment sensor network consisting of 1,170 smart sensors installed throughout Seoul city, providing various environmental information such as wind speed, wind direction, temperature, humidity, vibration, and pollutant concentration reported to one decimal place. All sensors (7 in total) located within the case study area were included in the analysis to prevent bias from selectively including or excluding specific sensors (refer to Table 6 and Fig. 4). The meteorological conditions for validation were selected based on two criteria: (i) sections where wind direction fluctuations remained stable to align with the steady-state assumption of flow analysis; (ii) wind speed levels capable of securing a sufficient signal-to-noise ratio compared to the sensor resolution (Blocken et al., 2013). To this end, based on the observation records of the Korea Meteorological Administration (KMA) automated weather station (AWS), the Mapo AWS located upstream of the target area, April 9, 2026, 10:00-12:00 KST was selected, and the predominant inflow wind direction and velocity for that section were defined as southeast and 3.1 m/s, respectively (KMA, 2020).

Next, the CFD analysis was performed using the commercial solver ANSYS Fluent 2021R1, and the turbulence model adopted the Reynolds-averaged Navier-Stokes (RANS)-based realizable k-ε model, which has a proven balance between prediction accuracy and computational cost in urban wind environment analysis (Blocken et al., 2016; Blocken, 2015). The computational domain was set to 8,858 m(x) × 8,858 m(y) × 987 m (z) applying a buffer length based on the COST 732 guideline to remove flow distortions (e.g., backflow, blockage effect) in the interest region (Franke et al., 2011; Tominaga et al., 2008). In setting boundary conditions, a no-slip condition and wall function were applied to the ground surface and building walls, a pressure outlet with a static pressure of 0 Pa was applied to the outlet, and a symmetry condition was applied to the

Table 5
Geometric fidelity between reference and modeled buildings.

Metric	Median	Within ±10%
Volume fidelity	0.994	99.8%
Footprint IoU	0.992	100% (>0.9)
Height accuracy	0 m	100% (<0.5 m)

Table 6
Information on the installation locations of the S-DoT sensors in Mapo-gu.

Sensor location	Sensor serial number	Latitude	Longitude
S1	OC3CL200254	37.55538	126.92137
S2	V02Q1940813	37.55462	126.89596
S3	OC3CL200244	37.56483	126.90245
S4	OC3CL200259	37.56231	126.91081
S5	OC3CL200246	37.54747	126.93325
S6	OC3CL200234	37.55192	126.93068
S7	OC3CL200242	37.55589	126.91037

top and sides of the domain. In particular, for the flow conditions of the inlet boundary, an atmospheric boundary layer (ABL) inlet profile was configured by applying the aforementioned observed wind speed and direction of the KMA Mapo AWS as reference values, and applying a power law exponent $\alpha = 0.28$ and aerodynamic roughness length $Z_0 = 1.0$ m considering urban morphological characteristics (Richards and Hoxey, 1993). A SIMPLE algorithm was applied for pressure-velocity coupling, and a second-order upwind scheme was applied for spatial discretization. The scope of the proposed pipeline ends with the generation of watertight surface geometry; the CFD volume mesh was generated in a separate downstream stage using the fault-tolerant meshing workflow of ANSYS Fluent, which produces a polyhedral-hexcore mesh directly from the exported surface geometry. The total number of grids in the computational domain, determined through grid convergence analysis, was 12,834,974 cells, essentially the same for the two geometries under the identical meshing settings. The above settings were equally applied to both the reference data and modeled data, and in both analyses, all convergence parameter residuals $<10^{-3}$ were achieved within 300 iterations.

The analysis result based on the reference data was calculated with an RMSE of 0.059 m/s ($R^2 = 0.994$) compared to the measured wind speed, confirming that the turbulence model and boundary condition settings adopted in this study reasonably reproduce the pedestrian-level wind environment of the target region (refer to Fig. 5). It should be noted that the S-DoT data are reported to one decimal place, so each measured value carries a quantization of ±0.05 m/s (Joint Committee for Guides in Metrology (JCGM), 2008). At five of the seven stations, the predicted wind speeds fell within this quantization interval of the measured values, indicating that the model predictions are indistinguishable from the measurements at the reporting precision of the data. The absolute error at the individual observation stations ranged from 0.009 m/s (S7) to 0.110 m/s (S1), corresponding to relative errors of 4.3–12.2% at the stations with measured wind speeds of 0.2 m/s or higher; at the three stations with a measured wind speed of 0.1 m/s, relative errors are dominated by the reporting quantization, and the corresponding absolute errors (0.009–0.033 m/s) are smaller than the ±0.05 m/s quantization interval. When evaluated using the hit-rate criterion of VDI 3783 standard (VDI 3783 Blatt 9, 2017), in which a predicted value is accepted when its absolute deviation from the measurement does not exceed the allowed absolute deviation W or its relative deviation does not exceed the allowed value of 25%, all seven stations were accepted, with W taken as the 0.05 m/s reporting precision of the data (hit rate of 100% against the required 66%); the hit rate remained 85.7% even under the relative-deviation criterion alone. While the absolute error at high-wind-speed stations (S1 to S4) was distributed in the range of 0.013 to 0.110 m/s, it appeared in the range of 0.009 to 0.033 m/s at low-wind-speed stations (S5 to S7), confirming that the flow stagnation characteristics of building-dense areas were also qualitatively reproduced. Meanwhile, the predicted wind speeds obtained from the reference and modeled geometries, computed as separate CFD runs under the same boundary conditions, were identical at all seven monitoring locations at the reported precision, with differences below 0.001 m/s, well within the solver convergence tolerance. These results suggest that the geometric simplifications introduced by the proposed workflow did not produce observable changes in the simulated wind field under the investigated CFD configuration, indicating that replacing the manual geometry-cleanup process with the proposed pipeline does not degrade the reliability of the analysis results.

4.3. Comparative evaluation on public benchmark datasets

To verify the practical applicability of the proposed pipeline beyond the heterogeneous GIS data of the case study, a comparative evaluation was performed on the public datasets associated with City4CFD, an open-source workflow explicitly addressing CFD-oriented urban geometry generation. City4CFD was selected as the reference tool because it targets the same application domain and reports validated results on two public campus datasets (Pađen et al., 2022, 2024). Because the exact building subsets of the published evaluation are not distributed, the evaluation was designed around the same public data sources and campus locations: for the TU Delft case, the campus dataset distributed with the City4CFD repository (AHN-derived building and ground point clouds); for the Stanford case, the USGS 3DEP point cloud of Santa Clara County (2020 collection; 23.6 pt/m^2, matching the 23.76 pt/m^2 of the published evaluation), with a circular domain of 1.6 km^2 centered on the Science and Engineering Quad, matching the published area size. The domain definitions therefore differ from those of the published evaluation, and the building counts reported below are not directly commensurate with the published figures; this difference is stated explicitly rather than concealed.

The evaluation uses the point-cloud input path of the proposed pipeline (Section 2.2): building identification by DBSCAN clustering of

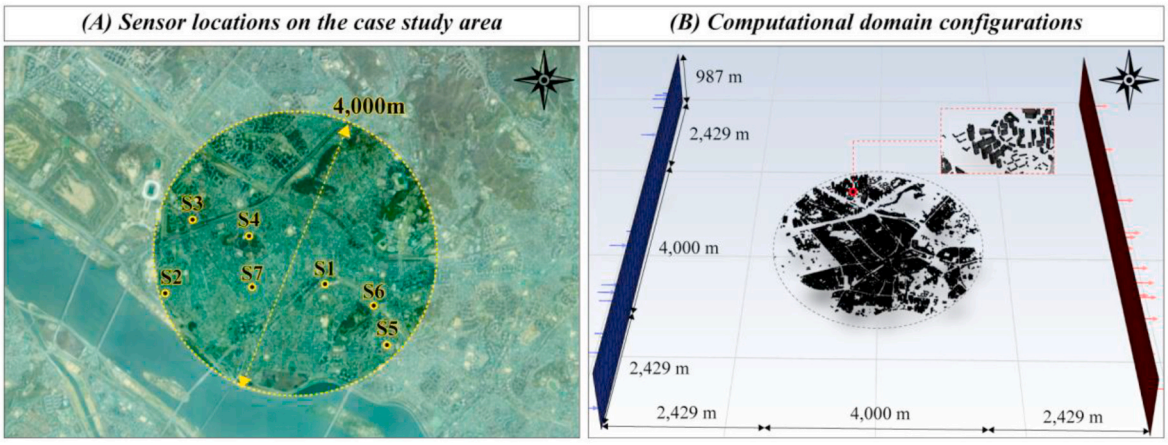


Fig. 4. Description of the sensor locations and CFD configurations in the case study area.

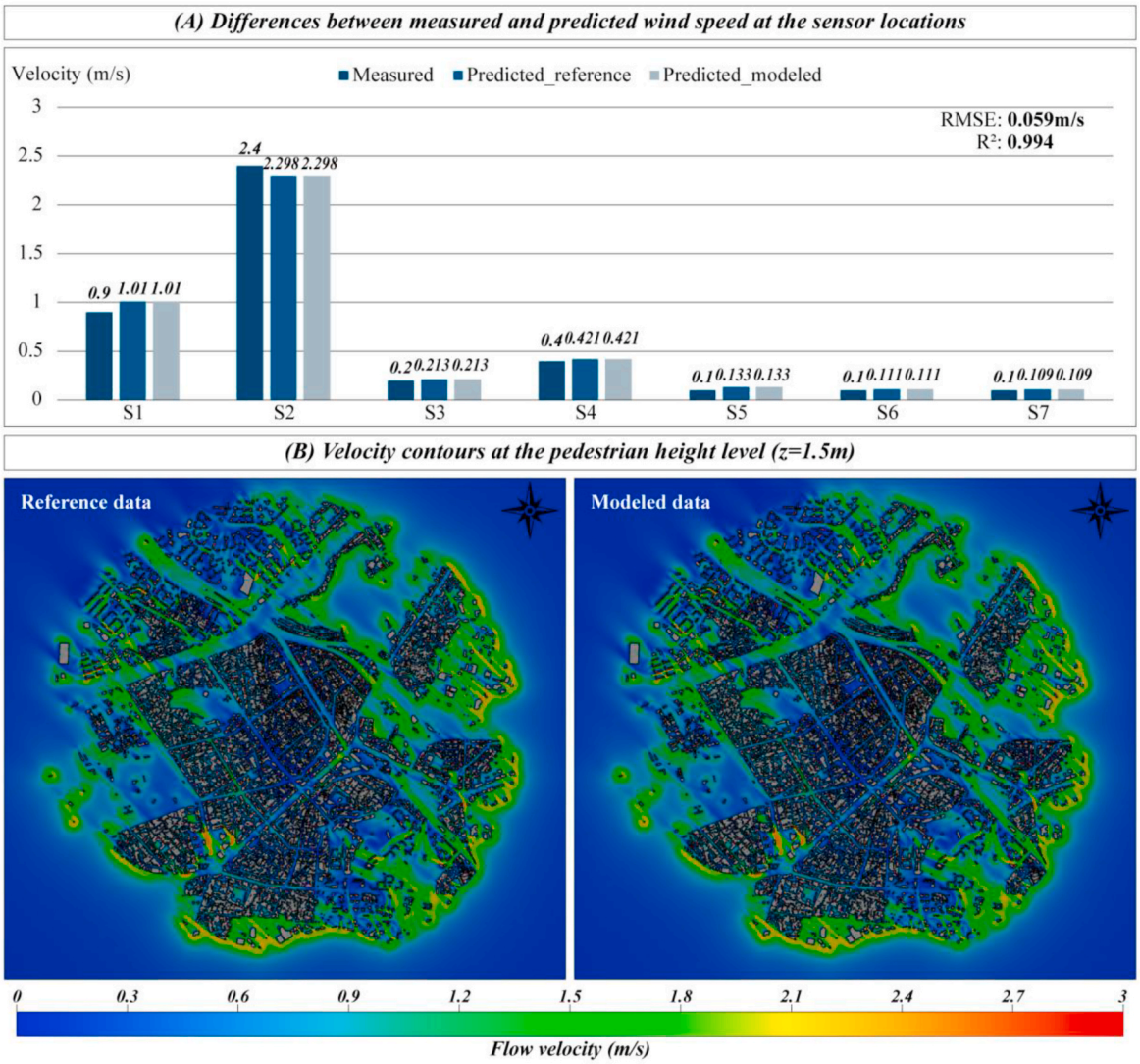


Fig. 5. CFD validation results.

the raw points, footprint extraction from the spatial point distribution, and height estimation from the point percentiles, all in the default configuration of Section 2.6 and without manual repair, parameter tuning, or post-hoc editing. Both the building and the ground point clouds were consumed: because classified airborne LiDAR building points are dominated by roof returns, the base elevation of each cluster was estimated as the median elevation of the ground points in the immediate vicinity of the cluster (Section 2.5), and the building height as the difference between the 99th-percentile roof elevation and this base elevation. The TU Delft clouds were used exactly as distributed in the City4CFD repository. For the Stanford dataset, the USGS 3DEP tiles were filtered to the building and ground classes (LAS classifications 6 and 2) and reprojected to metric coordinates—the same separation of building and ground points that the City4CFD preparation workflow requires.

For the TU Delft dataset, the point-cloud path identified 224 building clusters, of which 211 were reconstructed as target objects and 13 were excluded by the predefined analysis filters (13 below the minimum footprint area of 8 m^2, none below the minimum height); no reconstruction failed, and all 211 reconstructed buildings were watertight (8,732 faces), with no non-manifold edges and no degenerate faces. For the Stanford dataset, 184 clusters were identified, of which 181 were reconstructed and 3 were excluded by the filters (2 below the minimum height, 1 below the minimum footprint area); no reconstruction failed, and all 181 buildings were watertight (11,912 faces), with no non-manifold edges and no degenerate faces. The results are summarized in Table 7, and the reconstructed geometries are shown in Fig. 6.

Because the two workflows make different input assumptions, the following qualifications apply. First, City4CFD consumes cadastral building footprints (BAG, OpenStreetMap) together with the point clouds, whereas the proposed pipeline derives footprints from the point distribution itself; the density-based identification of the proposed pipeline identifies connected built masses rather than cadastral building parcels, merging physically attached structures (e.g., terraced row houses and connected campus wings) into single clusters. This object definition is suitable for LoD1 blockage representation in pedestrian-level CFD, but the resulting counts should not be interpreted as cadastral building counts. For the TU Delft dataset, 846 of the 984 distributed footprints that fall within the point-cloud extent contain at least 30 points, indicating that the smaller number of identified clusters reflects this merging semantics rather than missing point coverage. Second, City4CFD's published results report geometric validity against the ISO 19107 standard, with 4 of 76 (TU Delft) and 18 of 167 (Stanford) input buildings invalid prior to repair and 100% valid buildings after automatic alpha-wrap repair; the watertightness criterion used in this study (Section 2.7) is a necessary but not sufficient condition for ISO 19107 validity, and the two measures are presented side by side. Third,

Table 7
Comparative evaluation of the proposed pipeline and City4CFD on public campus datasets (TU Delft/Stanford).

Metric	City4CFD	Proposed pipeline
Input	Footprints (BAG/OSM) + building and ground point clouds	Building and ground point clouds only
Input preparation	76/167 (cadastral)	224/184 (DBSCAN, density-based)
Identification and input objects	70/167	211/181
Excluded by analysis filters	Not reported	13/3 ($H < 2.0$ m or $A < 8\,m^2$, Section 2.6)
Failed reconstructions	6/0	0/0
Geometric validity	4 of 76 and 18 of 167 invalid (pre-repair); 100% valid after alpha-wrap repair	100% watertight (211/211, 181/181)
Output LoD	LoD1.2–2.2	LoD1
Face count	Not reported	8,732/11,912
Residual micro-faces	Not reported	12/20 faces
Core pipeline execution	136 s/385 s (Threadripper 3970X, 24 cores)	26.7 s/33.6 s (virtualized server, 2 vCPUs, 2.9 GHz)
Manual intervention	None	None
Fidelity	Not reported	Not applicable (no reference geometry)

City4CFD reconstructs roof shapes up to LoD2.2, whereas the proposed pipeline produces LoD1 flat-roof geometry for pedestrian-level wind analysis; face-related metrics are reported as reference values. Fourth, fidelity metrics are not computed for these datasets, as no manually cleaned reference geometry exists. Fifth, the processing times are reported as contextual reference values rather than as a strict like-for-like runtime benchmark, since the exact published building subsets are not distributed and the LoD targets and object definitions differ. The hardware also differs: the present evaluation was executed on a virtualized server CPU (2 vCPUs, 2.9 GHz), which is substantially weaker than the 24-core workstation (AMD Threadripper 3970X) used in even under this setting, the core pipeline execution times of the proposed pipeline (26.7 s for TU Delft and 33.6 s for Stanford) were well below the corresponding runtimes reported for City4CFD (136 s and 385 s). City4CFD's authors separately note that the entire process from known data sources to final geometry took approximately 4 h including data download, preparation, and configuration, and that no manual modification of the reconstructed geometries was performed. The two workflows serve partly different simulation regimes: roof-shape detail (LoD2.2) increases the number of sharp edges and small faces and the associated mesh resolution requirements, which is practical at campus scale, whereas district-scale pedestrian-level wind analysis is conventionally performed with LoD1 geometry for computational feasibility, as demonstrated in the case study. The evaluation therefore verifies that the proposed pipeline remains reliable and efficient on the clean public data sources on which City4CFD was validated, while its distinct contribution—robustness to defective, heterogeneous GIS inputs without manual preprocessing—is demonstrated in the Mapo-gu case study (Section 4.1); the comparison characterizes complementary capabilities of the two workflows rather than the superiority of either.

5. Discussions

5.1. Repair versus reconstruction for CFD-oriented urban geometry preprocessing

The main distinction of the proposed approach lies in preprocessing urban CFD geometry through reconstruction rather than through the conventional repair of input defects. The repair functions of conventional manual workflows and commercial software fundamentally adopt an approach that locally corrects defects while preserving the topological connectivity of the input geometry. This approach is effective when individual defects are clearly and independently identifiable; however, when defects inherent in GIS data are complex or serially connected, cascade failure can occur where one correction induces new defects in adjacent regions. As observed in this study, the phenomenon where non-manifold edges increased after mesh clean-up application during the manual repair of the reference data empirically demonstrates the structural vulnerability of this repair approach. In contrast, the discretization-based reconstruction approach proposed in this study intentionally discards the topology of the input geometry and reconstructs the geometry from scratch based on the spatial distribution of point clouds. This approach applies the same process across a range of defect types and severities in the original data, thereby substantially reducing sensitivity to topological defects contained in the input geometry. However, this approach entails partial loss of original geometry information, determined by algorithmic approximation during the reconstruction process (e.g., α ratio, ε_{dp}). The geometric fidelity analysis (volume fidelity 0.994, footprint IoU 0.992) indicated that this information loss appears limited for the investigated pedestrian-level wind analysis application; however, for micro-level simulations where detailed building envelope geometry directly influences flow results (e.g., wind pressure distribution, natural ventilation analysis), this trade-off needs to be separately evaluated.

Furthermore, the reconstruction method of the proposed pipeline is fundamentally differentiated from existing geometry cleanup in that it performs geometry correction and physics-aware filtering simultaneously. In manual workflows, identifying and excluding non-building objects after geometry repair, and determining the geometry simplification level according to the analysis purpose, must be performed as separate manual tasks. In contrast, in this pipeline, such domain-specific judgments are internalized as physically justified parameters ($H_{\min}$, $A_{fp,\,min}$, ε_{dp}), so geometry reconstruction and physics-aware filtering are integrated and performed within a single process. The fact that the number of components in the modeled data (15,056) is slightly fewer than the reference data (15,072) is the result of small-scale structures that survived unidentified as non-buildings in manual cleanup being additionally excluded by physics-aware filtering, which suggests that rule-based parameters can capture non-building objects with a more consistent standard than manual identification by engineers.

5.2. Implications for downstream mesh generation

Differences in geometry quality directly affect not only the accuracy of the geometry itself but also the stability and efficiency of subsequent mesh generation processes. The micro-face ratio defined and computed in this study quantitatively reflects such downstream effects. The reduction from 12.2% in baseline data to 1.2% in modeled data goes beyond simple face count reduction, contributing to overall improvement of mesh quality (skewness, aspect ratio, size ratio) by substantially reducing the frequency at which extremely small cells are forcibly generated when the mesh generator discretizes surfaces. Notably, the micro-face ratio of modeled data is considerably lower than that of reference data (4.9%), indicating that boundary straightening through DP simplification provides a level of surface regularity that is difficult to achieve through commercial software-based manual cleanup.

In practice, during volume mesh generation using the fault-tolerant meshing workflow of ANSYS Fluent based on the reference data in this case study, numerous skewed or degenerate cells were generated in extremely small gap regions between adjacent buildings, requiring additional manual adjustments. This experience demonstrates the cascade effect where fine defects at the geometry stage propagate to the mesh generation stage, degrading the efficiency of the entire workflow. In contrast, the pipeline output maintains numerically distinct building geometries with minimized micro-faces, thereby structurally reducing the frequency of such adverse downstream effects.



Fig. 6. LoD1 geometries of the TU Delft (top) and Stanford (bottom) campus datasets.

Furthermore, the output format of the proposed pipeline—watertight STL mesh—corresponds to a standard format directly accepted as input by the mesh generation tools of major CFD solvers including ANSYS Fluent, OpenFOAM, and Star-CCM+. This means that the pipeline output is not tied to any specific commercial solver and can be immediately connected to diverse existing practical workflows without additional format conversion. In particular, end-to-end automation from geometry reconstruction to mesh generation is achievable through integration with SnappyHexMesh, the mesh generation module of the open-source CFD software OpenFOAM, which can serve as the foundation for an expanded pipeline including mesh generation automation in future research.

5.3. Scope and limitations

The pipeline of this study was designed for the purpose of pedestrian-level wind environment analysis, and accordingly, the LOD1 level geometry expression and physics-aware filtering parameters are optimized for that analysis purpose. These design choices entail the following limitations on the application scope.

First, the LOD1 expression does not include roof shapes, detailed elements of the envelope (e.g., balconies, canopies), and connecting structures between buildings (e.g., skybridges). When these elements significantly affect the analysis results (e.g., analyzing flow fields on building rooftops, calculating wind pressure distribution on the building envelope, or evaluating natural ventilation performance), geometry expressions of LOD2 or higher are required. However, the concave hull-based footprint extraction and extrusion structure of this pipeline are structurally capable of expanding to LOD2 by adding a roof shape classification module (e.g., determining roof types from LiDAR point clouds) to the back end, which is a major expansion direction for future research.

Second, the hyperparameters applied in this study (e.g., α ratio, ε_{dp}, $H_{\min}$, $A_{fp,min}$) were set based on the urban morphological characteristics of Mapo-gu, Seoul city. In urban environments where the form, density, and separation distance between buildings differ greatly from the target site of this study (e.g., ultra-high-density downtowns, low-density sub-urban areas, historical city centers), parameter readjustment may be required. These parameters are designed as user-defined variables within the pipeline, making adjustment itself easy, yet parameter sensitivity analysis for various urban types needs to be systematically performed in future research.

Third, the validation of this study was performed for a district-level region containing approximately 15,000 building components. While the pipeline's processing time demonstrated high efficiency at this scale, scalability for large-scale batch processing at the macro-urban level involving hundreds of thousands of buildings or multiple cities has not been

separately verified. However, since each stage of the pipeline operates independently on a building-by-building basis, extension through parallel processing is feasible in principle.

6. Conclusions

In this study, a discretization-based reconstruction pipeline that automatically generates watertight LOD1 building geometry from raw GIS data was proposed, and empirical validation was performed on a dense urban region containing approximately 15,000 buildings.

The proposed pipeline achieved equal or superior results to multi-stage manual cleanup across all geometric quality metrics. Specifically, it secured 100% watertightness, 0 non-manifold edges, and 0 degenerate face buildings, and the proportion of micro-face buildings decreased to 1.2% compared to 4.9% for manual cleanup. In terms of geometric fidelity, it achieved a volume fidelity of 0.994, footprint IoU of 0.992, and height accuracy of 0 m compared to the reference data, confirming that the information loss accompanying geometry reconstruction is negligible in pedestrian-level wind analysis. Furthermore, the total processing time required for geometry preprocessing was 190 s for approximately 15,000 buildings on a single workstation, reducing an effort that requires manual work on the order of several days to a matter of minutes.

These results carry the following academic and practical implications. First, by reconstructing geometry from sampled points rather than repairing input topology directly, the workflow reduces the cascade failure risk inherent in conventional repair approaches and improves the robustness of automated preprocessing across defective GIS inputs. Second, as physics-aware filtering previously dependent on engineers' empirical judgment was internalized as quantified parameters, reproducibility of preprocessing results was secured. Guaranteeing identical output for identical input is a characteristic unattainable in manual workflows, which can in turn enhance the comparability of urban CFD research. Third, through the integration of geometry reconstruction and physics-aware filtering into a single process, the preprocessing effort previously dependent on expert manual work was substantially reduced, lowering the barrier for a broader range of researchers to access urban CFD analysis. This can contribute to the expansion of urban wind environment assessment from the domain of a few specialists to diverse fields including urban planning, architectural design, and environmental policy formulation for sustainable urban development. In future research, the scope of the pipeline can be expanded through extension to LOD2, integration with automated mesh generation, and parameter applicability verification for various urban types.

Beyond geometry preprocessing, the proposed workflow may support large-scale generation of CFD-ready urban datasets for emerging applications such as surrogate modeling, reduced-order modeling, and machine-learning-assisted urban airflow prediction. In this sense, the framework provides infrastructure for scalable dataset generation in addition to geometry preparation.

CRediT authorship contribution statement

Jinwoo Choi: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Taehoon Hong:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **Hyeonggyun Kim:** Writing – original draft, Visualization, Validation, Resources, Formal analysis. **Kwangbok Jeong:** Writing – review & editing, Validation, Supervision, Project administration, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT; Ministry of Science and ICT) (NRF-2021R1A3B1076769).

Data availability

The source code, the configuration used in the study, the measurement and table-reproduction scripts, and the baseline, reference, and modeled geometry files of the case study are publicly available at https://github.com/jinwoo818/Discretization-based-reconstruction-of-raw-GIS-to-watertight-geometry. The original GIS source data are subject to data-provider licensing and are available from the corresponding author upon reasonable request.

References

Abu-Zidan, Y., Mendis, P., Gunawardena, T., 2021. Optimising the computational domain size in CFD simulations of tall buildings. Heliyon 7, e06723. https://doi.org/10.1016/j.heliyon.2021.e06723.

AbuBaker, A., Qahwaji, R., Ipson, S., Saleh, M., 2007. One scan connected component labeling technique. In: 2007 IEEE Int. Conf. Signal Process. Commun., pp. 1283–1286. https://doi.org/10.1109/ICSPC.2007.4728561. IEEE, Dubai, United Arab Emirates.

Ali, Z., Tyacke, J., Watson, R., Tucker, P.G., Shahpar, S., 2019. Efficient preprocessing of complex geometries for CFD simulations. Int. J. Comput. Fluid Dynam. 33, 98–114. https://doi.org/10.1080/10618562.2019.1606421.

Ali Smail, S., Zemmouri, N., Djenane, M., Nikolopoulou, M., 2024. Investigating the transient conditions of "Sabat" space and its influence on pedestrian sensations during thermal walks. Algiers' casbah case study. Build. Environ. 261, 111760. https://doi.org/10.1016/j.buildenv.2024.111760.

ANSYS, 2017. Best Practice: RANS Turbulence Modeling in Ansys CFD. https://www.ansys.com/content/dam/amp/2022/february/asset-creation/best-practices-campaign/BestPractice-RansturbulencemodelinginAnsysCFD.pdf. (Accessed 30 August 2023).

ANSYS, 2020. Ansys fluent 2020 R1-Theory guide. https://ansyshelp.ansys.com/account/secured?returnurl=/Views/Secured/corp/v201/en/flu_th/flu_th.html?q=ansysfluenttheoryguide%0A%0A. February 23, 2023.

Asaeedi, S., Didehvar, F., Mohades, A., 2017. α-Concave hull, a generalization of convex hull. Theor. Comput. Sci. 702, 48–59. https://doi.org/10.1016/j.tcs.2017.08.014.

Baitureyeva, A., Yang, T., Wang, H.S., 2025. Development of machine learning-aided rapid CFD prediction for optimal urban wind environment design. Sustain. Cities Soc. 121, 106208. https://doi.org/10.1016/j.scs.2025.106208.

Barber, C.B., Dobkin, D.P., Huhdanpaa, H., 1996. The quickhull algorithm for convex hulls. ACM Trans. Math Software 22, 469–483. https://doi.org/10.1145/235815.235821.

Becker, S., 2009. Generation and application of rules for quality dependent façade reconstruction. ISPRS J. Photogrammetry Remote Sens. 64, 640–653. https://doi.org/10.1016/j.isprsjprs.2009.06.002.

Beil, C., Kolbe, T.H., 2017. CITYGML and the streets of NEW YORK -a proposal for detailed street space modelling. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci. IV-4/W5, 9–16. https://doi.org/10.5194/isprs-annals-IV-4-W5-9-2017.

Biljecki, F., Ledoux, H., Stoter, J., 2016a. An improved LOD specification for 3D building models. Comput. Environ. Urban Syst. 59, 25–37. https://doi.org/10.1016/j.compenvurbsys.2016.04.005.

Biljecki, F., Ledoux, H., Stoter, J., 2017. Generating 3D city models without elevation data. Comput. Environ. Urban Syst. 64, 1–18. https://doi.org/10.1016/j.compenvurbsys.2017.01.001.

Biljecki, F., Ledoux, H., Stoter, J., Vosselman, G., 2016b. The variants of an LOD of a 3D building model and their influence on spatial analyses. ISPRS J. Photogrammetry Remote Sens. 116, 42–54. https://doi.org/10.1016/j.isprsjprs.2016.03.003.

Biljecki, F., Ledoux, H., Stoter, J., Zhao, J., 2014. Formalisation of the level of detail in 3D city modelling. Comput. Environ. Urban Syst. 48, 1–15. https://doi.org/10.1016/j.compenvurbsys.2014.05.004.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D city models: state of the art review. ISPRS Int. J. GeoInf. 4, 2842–2889. https://doi.org/10.3390/ijgi4042842.

Blocken, B., 2015. Computational fluid dynamics for urban physics: importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations. Build. Environ. 91, 219–245. https://doi.org/10.1016/j.buildenv.2015.02.015.

Blocken, B., Stathopoulos, T., van Beeck, J.P.A.J., 2016. Pedestrian-level wind conditions around buildings: review of wind-tunnel and CFD techniques and their accuracy for wind comfort assessment. Build. Environ. 100, 50–81. https://doi.org/10.1016/j.buildenv.2016.02.004.

Blocken, B., Tominaga, Y., Stathopoulos, T., 2013. CFD simulation of micro-scale pollutant dispersion in the built environment. Build. Environ. 64, 225–230. https://doi.org/10.1016/j.buildenv.2013.01.001.

Blocken, B., van der Hout, A., Dekker, J., Weiler, O., 2015. CFD simulation of wind flow over natural complex terrain: case study with validation by field measurements for

Ria de Ferrol, Galicia, Spain. J. Wind Eng. Ind. Aerod. 147, 43–57. https://doi.org/10.1016/j.jweia.2015.09.007.
Brunekreef, B., Holgate, S.T., 2002. Air pollution and health. Lancet 360, 1233–1242. https://doi.org/10.1016/S0140-6736(02)11274-8.
Campello, R.J.G.B., Moulavi, D., Sander, J., 2013. Density-based clustering based on hierarchical density estimates. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (Eds.), Adv. Knowl. Discov. Data Min. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 160–172. https://doi.org/10.1007/978-3-642-37456-2_14.
Chajaei, F., Bagheri, H., 2025. LOD1 3D city model from LiDAR: the impact of segmentation accuracy on quality of urban 3D modeling and morphology extraction. Remote Sens. Appl.: Soc. Environ. 38, 101534. https://doi.org/10.1016/j.rsase.2025.101534.
Choi, J., Hong, J., Kang, H., Hong, T., Park, H.S., Lee, D.-E., 2022. An automatic decision model for optimal noise barrier plan in terms of health impact, productivity, and cost aspects. Build. Environ. 216, 109033. https://doi.org/10.1016/j.buildenv.2022.109033.
Deininger, M.E., Von Der Grün, M., Piepereit, R., Schneider, S., Santhanavanich, T., Coors, V., Voß, U., 2020. A continuous, semi-automated workflow: from 3D city models with geometric optimization and CFD simulations to visualization of wind in an urban environment. ISPRS Int. J. GeoInf. 9, 657. https://doi.org/10.3390/ijgi9110657.
Dillencourt, M.B., Samet, H., Tamminen, M., 1992. A general approach to connected-component labeling for arbitrary image representations. J. ACM 39, 253–280. https://doi.org/10.1145/128749.128750.
Ding, S., Huang, Y., Cui, P., Wu, J., Li, M., Liu, D., 2019. Impact of viaduct on flow reversion and pollutant dispersion in 2D urban street canyon with different roof shapes - numerical simulation and wind tunnel experiment. Sci. Total Environ. 671, 976–991. https://doi.org/10.1016/j.scitotenv.2019.03.391.
Dorninger, P., Pfeifer, N., 2008. A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. Sensors 8, 7323–7343. https://doi.org/10.3390/s8117323.
Douglas, D.H., Peucker, T.K., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica 10, 112–122. https://doi.org/10.3138/FM57-6770-U75U-7727.
Edelsbrunner, H., Kirkpatrick, D., Seidel, R., 1983. On the shape of a set of points in the plane. IEEE Trans. Inf. Theor. 29, 551–559. https://doi.org/10.1109/TIT.1983.1056714.
Fishkin, K.P., Barsky, B.A., 1985. An analysis and algorithm for filling propagation. In: Magnenat-Thalmann, N., Thalmann, D. (Eds.), Comput.-Gener. Images. Springer Japan, Tokyo, pp. 56–76. https://doi.org/10.1007/978-4-431-68033-8_6.
Franke, J., Hellsten, A., Schlunzen, K.H., Carissimo, B., 2011. The COST 732 best practice guideline for CFD simulation of flows in the urban environment: a summary. Int. J. Environ. Pollut. 44, 419. https://doi.org/10.1504/IJEP.2011.038443.
Haala, N., Kada, M., 2010. An update on automatic 3D building reconstruction. ISPRS J. Photogrammetry Remote Sens. 65, 570–580. https://doi.org/10.1016/j.isprsjprs.2010.09.006.
He, Y., Liu, X.-H., Zhang, H.-L., Zheng, W., Zhao, F.-Y., Aurel Schnabel, M., Mei, Y., 2021. Hybrid framework for rapid evaluation of wind environment around buildings through parametric design, CFD simulation, image processing and machine learning. Sustain. Cities Soc. 73, 103092. https://doi.org/10.1016/j.scs.2021.103092.
Joint Committee for Guides in Metrology (JCGM), 2008. Evaluation of measurement data — guide to the expression of uncertainty in measurement. JCGM 100, 2008. https://www.bipm.org/documents/20126/2071204/JCGM_100_2008_E.pdf/cb0ef43f-baa5-11cf-3f85-4dcd86f77bd6. September 15, 2026.
Karkoulias, V.A., Marazioti, P.E., Georgiou, D.P., Maraziotis, E.A., 2020. Computational fluid dynamics modeling of the trace elements dispersion and comparison with measurements in a street canyon with balconies in the city of Patras, Greece, atmos. Environ. Times 223, 117210. https://doi.org/10.1016/j.atmosenv.2019.117210.
Kastner, P., Dogan, T., 2023. A GAN-based surrogate model for instantaneous urban wind flow prediction. Build. Environ. 242, 110384. https://doi.org/10.1016/j.buildenv.2023.110384.
KMA, 2020. Meteorological data open portal, Seoul South Korea. https://data.kma.go.kr/cmmn/main.do. (Accessed 10 April 2026).
Kolbe, T.H., Gröger, G., Plümer, L., 2005. CityGML: interoperable access to 3D city models. In: Van Oosterom, P., Zlatanova, S., Fendel, E.M. (Eds.), Geo-Inf. Disaster Manag. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 883–899. https://doi.org/10.1007/3-540-27468-5_63.
Korea Land and Housing Corporation, 2018. Open data portals of land and real estates. Gyeongsangnam- South Korea. https://seereal.lh.or.kr/. (Accessed 1 February 2020).
Ledoux, H., Biljecki, F., Dukai, B., Kumar, K., Peters, R., Stoter, J., Commandeur, T., 2021. 3dfier: automatic reconstruction of 3D city models. J. Open Source Softw. 6, 2866. https://doi.org/10.21105/joss.02866.
Ledoux, H., Meijers, M., 2011. Topologically consistent 3D city models obtained by extrusion. Int. J. Geogr. Inf. Sci. 25, 557–574. https://doi.org/10.1080/13658811003623277.
Linda, J., Pospíšil, J., Köbölová, K., 2022. Identification of wind-induced particle resuspension in urban environment using CFD modelling. Atmosphere 14, 57. https://doi.org/10.3390/atmos14010057.
Liu, S., Pan, W., Zhang, H., Cheng, X., Long, Z., Chen, Q., 2017. CFD simulations of wind distribution in an urban community with a full-scale geometrical model. Build. Environ. 117, 11–23. https://doi.org/10.1016/j.buildenv.2017.02.021.
MOLIT, 2018. Enforcement decree on the building act, Sejong city South Korea. https://elaw.klri.re.kr/kor_service/lawView.do?hseq=46640&lang=ENG. (Accessed 2 December 2019).
MOLIT, 2021. Korea National Spatial Data Infrastructure Portal. http://www.nsdi.go.kr/lxportal/?menuno=2679. (Accessed 30 September 2021).
Musialski, P., Wonka, P., Aliaga, D.G., Wimmer, M., Van Gool, L., Purgathofer, W., 2013. A survey of urban reconstruction. Comput. Graph. Forum 32, 146–177. https://doi.org/10.1111/cgf.12077.
Pađen, I., García-Sánchez, C., Ledoux, H., 2022. Towards automatic reconstruction of 3D city models tailored for urban flow simulations. Front. Built Environ. 8, 899332. https://doi.org/10.3389/fbuil.2022.899332.
Pađen, I., Peters, R., García-Sánchez, C., Ledoux, H., 2024. Automatic high-detailed building reconstruction workflow for urban microscale simulations. Build. Environ. 265, 111978. https://doi.org/10.1016/j.buildenv.2024.111978.
Park, G., Kim, C., Lee, M., Choi, C., 2020. Building geometry simplification for improving mesh quality of numerical analysis model. Appl. Sci. 10, 5425. https://doi.org/10.3390/app10165425.
Peters, R., Dukai, B., Vitalis, S., Van Liempt, J., Stoter, J., 2022. Automated 3D reconstruction of LoD2 and LoD1 models for all 10 million buildings of the Netherlands, photogramm. Eng. Remote Sens. 88, 165–170. https://doi.org/10.14358/PERS.21-00032R2.
Piepereit, R., Deininger, M., Kada, M., Pries, M., Voß, U., 2018. A sweep-plane algorithm for the simplification of 3D building models in the application scenario of wind simulations. Int. Arch. Photogram. Rem. Sens. Spatial Inf. Sci. XLII-4/W10, 151–156. https://doi.org/10.5194/isprs-archives-XLII-4-W10-151-2018.
Richards, P.J., Hoxey, R.P., 1993. Appropriate boundary conditions for computational wind engineering models using the k-ϵ turbulence model. J. Wind Eng. Ind. Aerod. 46–47, 145–153. https://doi.org/10.1016/0167-6105(93)90124-7.
Rottensteiner, F., 2003. Automatic generation of high-quality building models from lidar data. IEEE Comput. Graph. Appl. 23, 42–50. https://doi.org/10.1109/MCG.2003.1242381.
Sanyal, P., 2023. AELH-, CFD-, and ANN-Based wind interference zone prediction of regular tall buildings. Asian J. Civ. Eng. 24, 3881–3891. https://doi.org/10.1007/s42107-023-00683-5.
Schubert, E., Sander, J., Ester, M., Kriegel, H.P., Xu, X., 2017. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. ACM Trans. Database Syst. 42, 1–21. https://doi.org/10.1145/3068335.
Seoul Metropolitan Government, 2023. Seoul Open Data Plaza White Paper. https://data.seoul.go.kr/asset/downloadFiles/opendatabook_e.pdf. (Accessed 30 March 2026).
Seoul Metropolitan Government, 2026. Smart Seoul city data sensor (S-DoT) environmental information (Real-time). https://data.seoul.go.kr/dataList/OA-22833/A/1/datasetView.do.
Shi, W., Cheung, C., 2006. Performance evaluation of line simplification algorithms for vector generalization. Cartogr. J. 43, 27–44. https://doi.org/10.1179/000870406X93490.
Shi, Y., Yang, R., Wu, Z., Li, P., Liu, C., Zhao, H., Zhou, G., 2024. City-scale continual neural semantic mapping with three-layer sampling and panoptic representation. Knowl.-Based Syst. 284, 111145. https://doi.org/10.1016/j.knosys.2023.111145.
Tominaga, Y., Mochida, A., Yoshie, R., Kataoka, H., Nozu, T., Yoshikawa, M., Shirasawa, T., 2008. AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings. J. Wind Eng. Ind. Aerod. 96, 1749–1761. https://doi.org/10.1016/j.jweia.2008.02.058.
Van Den Brink, L., Stoter, J., Zlatanova, S., 2013. Establishing a national standard for 3D topographic data compliant to CityGML. Int. J. Geogr. Inf. Sci. 27, 92–113. https://doi.org/10.1080/13658816.2012.667105.
Vardoulakis, S., Fisher, B.E.A., Pericleous, K., Gonzalez-Flesca, N., 2003. Modelling air quality in street canyons: a review, atmos. Environ. Times 37, 155–182. https://doi.org/10.1016/S1352-2310(02)00857-9.
VDI 3783 blatt 9 - environmental meteorology - prognostic microscale wind field models - evaluation for flow around buildings and obstacles. https://www.vdi.de/en/home/vdi-standards/details/vdi-3783-blatt-9-environmental-meteorology-prognostic-microscale-wind-field-models-evaluation-for-flow-around-buildings-and-obstacles, 2017. September 15, 2026.
Zhang, H., Zou, Y., Yan, Z., Cheng, H., 2024. Rapid-mapping: lidar-visual implicit neural representations for real-time dense mapping. IEEE Rob. Autom. Lett. 9, 8154–8161. https://doi.org/10.1109/LRA.2024.3440729.
Zhang, X., Yu, R., Ren, S., 2025. Neural implicit representations for multi-view surface reconstruction: a survey. IEEE Trans. Vis. Comput. Graph. 31, 9444–9463. https://doi.org/10.1109/TVCG.2025.3582627.
Zhong, J., Cai, X.-M., Bloss, W.J., 2017. Large eddy simulation of reactive pollutants in a deep urban street canyon: coupling dynamics with O3-NO -VOC chemistry. Environ. Pollut. 224, 171–184. https://doi.org/10.1016/j.envpol.2017.01.076.