buildings

MDPI

Systematic Review

Toward a Unified Smart Point Cloud Framework: A Systematic Review of Definitions, Methods, and a Modular Knowledge-Integrated Pipeline

Mohamed H. Salaheldin *, Ahmed Shaker and Songnian Li

Department of Civil Engineering, Toronto Metropolitan University, 350 Victoria Street, Toronto, ON M5B 2K3, Canada; ahmed.shaker@torontomu.ca (A.S.); snli@torontomu.ca (S.L.)
* Correspondence: mohamed.h.ismail@torontomu.ca

Abstract

Reality-capture has made point clouds a primary spatial data source, yet processing and integration limits hinder their potential. Prior reviews focus on isolated phases; by contrast, Smart Point Clouds (SPCs)—augmenting points with semantics, relations, and query interfaces to enable reasoning—received limited attention. This systematic review synthesizes the state-of-the-art SPC terminology and methods to propose a modular pipeline. Following PRISMA, we searched Scopus, Web of Science, and Google Scholar up to June 2025. We included English-language studies in geomatics and engineering presenting novel SPC methods. Fifty-eight publications met eligibility criteria: *Direct* (n = 22), *Indirect* (n = 22), and *New Use* (n = 14). We formalize an operative SPC definition—queryable, ontology-linked, provenance-aware—and map contributions across traditional point cloud processing stages (from acquisition to modeling). Evidence shows practical value in cultural heritage, urban planning, and AEC/FM via semantic queries, rule checks, and auditable updates. Comparative qualitative analysis reveals cross-study trends: higher and more uniform density stabilizes features but increases computation, and hybrid neuro-symbolic classification improves long-tail consistency; however, methodological heterogeneity precluded quantitative synthesis. We distill a configurable eight-module pipeline and identify open challenges in data at scale, domain transfer, temporal (4D) updates, surface exports, query usability, and sensor fusion. Finally, we recommend lightweight reporting standards to improve discoverability and reuse.

Keywords: smart point cloud; rich point cloud; scan-to-BIM; advanced point cloud processing; geometric reasoning; systematic literature review

Check for updates

Academic Editor: Byungjoo Choi

Received: 21 November 2025
Revised: 3 January 2026
Accepted: 6 January 2026
Published: 10 January 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

A point cloud is a set of three-dimensional (3D) samples that captures the geometry of objects or environments [1]. In remote sensing, point clouds are generated via surveying techniques to model natural landscapes and the built environment. Each point is specified by Cartesian coordinates (x, y, z) and may carry additional attributes depending on the sensing system—e.g., color, surface normal, intensity, number of returns, GPS time, and scan angle [2]. Although often treated as intermediate products, point clouds are central to documenting as-built conditions and to enabling downstream modeling and analysis because of their spatial precision and geometric richness [3].

The growing adoption of reality-capture technologies has made point clouds a primary source of spatial data. Common acquisition platforms include active sensors such as Light

Detection and Ranging (LiDAR)—used in airborne laser scanning (ALS), terrestrial laser scanning (TLS), and unmanned aerial vehicles (UAVs)—as well as other depth-sensing devices. These modalities support a broad range of applications spanning design and construction [4], topographic mapping [5], environmental monitoring [6], infrastructure development [7], and cultural heritage preservation [8]. More recently, passive imaging systems have also been utilized to generate point clouds through photogrammetry and structure-from-motion (SfM) techniques using Red-Green-Blue (RGB), infrared, or thermal imagery [9]. With increased accessibility and lower costs, these tools enable rapid, high-resolution data acquisition [10]. However, raw point clouds frequently contain noise and outliers, exhibit occlusions and irregular sampling density, and produce huge files—complicating storage, processing, and data exchange [11].

Point cloud processing transforms raw data into structured, information-rich representations through algorithmic workflows and data integration. Processed outputs typically incorporate attributes that facilitate analyses and data exchange. Beyond purely geometric operations, many tasks require integrating domain semantics so that objects and relations are machine-interpretable. Semantically enriched point clouds enable automated reasoning and task-specific applications, such as scene interpretation [12]. However, given the high dimensionality of point clouds, processing demands complex, automated methods capable of interpreting geometric structure and semantic context.

The Smart Point Cloud (SPC) concept has emerged to address these challenges by embedding intelligence into point cloud workflows: a semantically structured point cloud model that encodes domain knowledge and supports machine-interpretable reasoning, automation, and decision support [13]. Early SPC work largely extended conventional pipelines—using geometric and spectral features for classification, followed by semantic labeling to integrate knowledge. For example, a web-based prototype on heritage tesserae stones demonstrated interactive streaming with real-time semantic annotation, signaling the first shift toward SPC ontology-based interpretation while remaining close to traditional workflows [14]. Since then, efforts have broadened SPC beyond static classification toward lifecycle-aware systems that support incremental updates, context-aware interpretation, and links to external knowledge bases (e.g., information models or Digital Twins (DTs). In these settings, point clouds evolve into continuously updated digital assets that participate in analysis, simulation, and decision-making across capture, construction/operation, and maintenance phases.

Figure 1 summarizes three pillars that enable this progression: Knowledge Extraction (KE), Knowledge Integration (KI), and Knowledge Representation (KR), which together address challenges in structuring, linking, and interpreting large, multi-source datasets [15]. KE involves data preparation, Data Mining (DM) tasks for information, and developing new knowledge from data [16]—feature extraction, segmentation, clustering, and classification [17]. KI synthesizes and aligns the multiple knowledge sources, understanding, representing, and layering them by domains. KR then formalizes this knowledge—e.g., ontologies and rules expressed in machine-readable languages—to enable constraint checking, logical inference, and workflow automation with reduced human input [2]. The decision-making and inference functions that consume these KR fall in the Cognitive Decision Layer (CDL).

Despite progress, SPC remains a fragmented research area, with inconsistencies in terminology, data sources, and processing methods. To the best of our knowledge, no recent Systematic Literature Review (SLR) has comprehensively synthesized developments in this domain. This study addresses this gap through a rigorous SLR aimed at unifying knowledge and guiding future research. The objectives of this review are to (i) identify and categorize SPC-focused studies; (ii) evaluate consistency in terminology, definitions,

and conceptual frameworks; (iii) extract and compare data types, processing methods, and workflows; and (iv) identify open challenges and propose future research directions. In response to these objectives, this study provides the following:

1. An evidence-backed overview of SPC workflows situated within traditional point cloud processing, covering data types, pre-processing, and semantic integration;
2. A comparative taxonomy of SPC definitions, methodologies, datasets, and tools across application domains;
3. A prioritized agenda of SPCs' unresolved challenges and opportunities to guide future innovation in remote sensing and spatial data science.

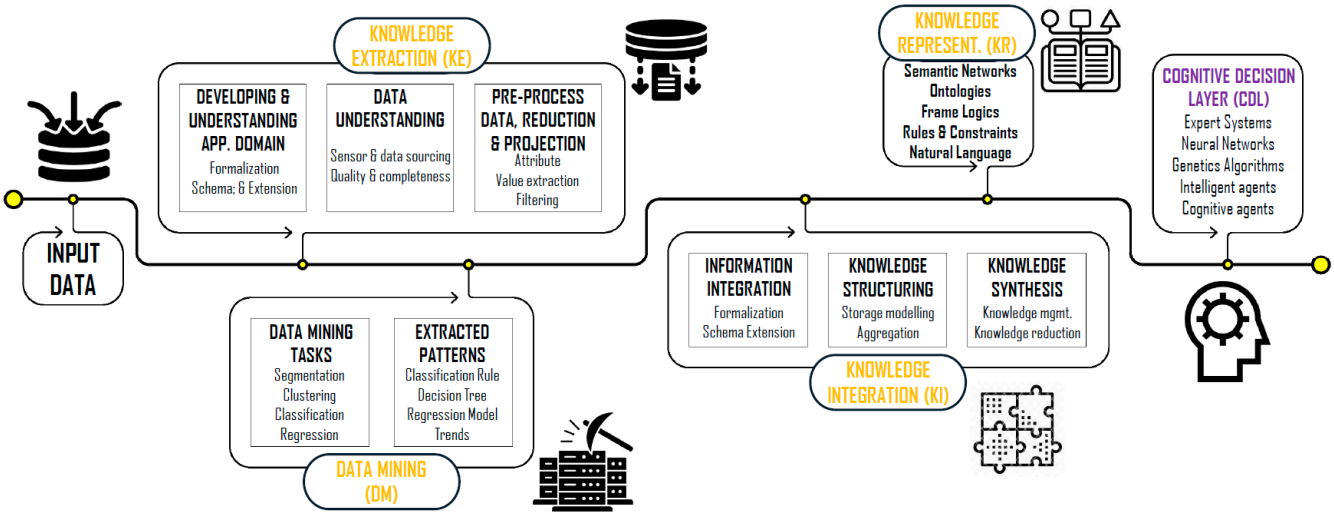


Figure 1. Conceptual framework for developing intelligent virtual environments, illustrating the transition from Knowledge Extraction to automated reasoning via the Cognitive Decision Layer (CDL)—Adapted from [15].

The remainder of the paper is organized as follows. Section 2 introduces traditional point cloud workflows and positions SPC in context. Section 3 details the SLR methodology. Section 4 presents the review's synthesis and findings. Section 5 discusses the results in light of the research questions. Section 6 outlines open challenges and future research directions. Section 7 concludes the paper.

2. Traditional Point Cloud Processing and SPC Shift

Point cloud processing is foundational in remote sensing as datasets grow denser and more complex [18]. Typically performed at the point level, it mitigates noise, non-uniform density, and missing semantic context to improve geometric fidelity and downstream interpretability [19]. Despite extensive research, point cloud processing remains an ill-posed problem—solutions can be non-unique, sensitive to perturbations, or fail to converge [20].

Processing is integral in almost all scenarios. For example, it supports the integration of heterogeneous datasets, environment updates via change detection, and accuracy refinement through robust registration and filtering. With auxiliary data, it also enables semantic enhancement, transforming raw points into analyzable representations suitable for modeling, decision-making, and automation [21].

From a remote sensing standpoint, Che et al. [22] categorize core data processing tasks into four stages—feature extraction, segmentation, object recognition, and classification—a sequence that underpins semantic interpretation. In point cloud contexts, Abreu et al. [23] propose a pipeline that begins with data collection (including planning and attribute selection), followed by the processing steps such as classification and modeling. Effec-

tive deployment of these workflows also requires scoping application goals, selecting datasets/algorithms, and applying appropriate pre-processing [2]. To harmonize different views, we propose a generalized six-stage workflow for traditional point cloud processing, shown in Figure 2.

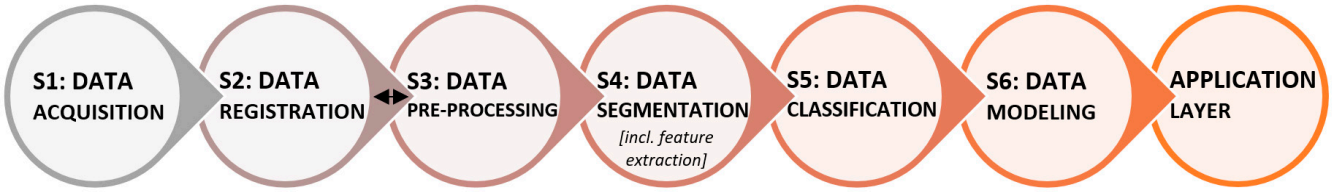


Figure 2. Generalized six-stage workflow for traditional point cloud processing, from acquisition through data modeling and application-specific structuring.

The workflow proceeds with *S1 Acquisition*, where point clouds are captured using sensors such as LiDAR or photogrammetric cameras. Collection parameters are also configured at this stage according to environmental and project-specific requirements (platform, trajectory, overlap, exposure). This is followed by *S2 Registration*, where multiple scans are aligned into a common frame using feature-based or Iterative Closest Point (ICP)-style methods, optionally constrained by Global Navigation Satellite System (GNSS)/Inertial Measurement Unit (IMU). *S3 Pre-processing* then improves data quality through operations such as outlier removal, denoising, subsampling, and error correction.

It should be noted that the relative ordering of *S2 Registration* and *S3 Pre-processing* depends on sensing modality and workflow design. For example, in ALS/Mobile Laser Scanner (MLS), an initial trajectory solution (e.g., GNSS/IMU, or Simultaneous Localization and Mapping (SLAM)) provides a first registration during acquisition; most pre-processing then occurs afterward in the unified frame; while in TLS, two patterns are common: (i) light per-scan pre-processing (e.g., outlier removal, subsampling) to ease overlap detection, followed by registration and then global pre-processing if needed; or (ii) register first, then apply global pre-processing. In this review, we discuss steps and algorithms by function, while acknowledging that, operationally, steps may interleave.

S4 Segmentation partitions the cloud into coherent regions, supported by *Feature Extraction*, which derives attributes such as curvature, eigen-features, local descriptors, or learned embeddings for pattern recognition. In *S5 Classification*, instances of segmented entities are localized (object detection/instance segmentation), and semantic labels are assigned using methods like rule-based systems, Machine Learning (ML) models, or manual annotation. Finally, *S6 Data Modeling* (and Structuring) maps labeled entities to application schemas (e.g., Industry Foundation Classes (IFC) for Building Information Modeling (BIM), City Geography Markup Language (CityGML) for urban models, domain taxonomies for heritage) and packages them for the *Application Layer*. In traditional workflows, these outputs feed application-specific analytics; in SPC settings, once formalized via *KR*, they enable *CDL* functions for automated reasoning.

While this traditional pipeline yields structured geometry, semantic depth and machine-interpretable knowledge are limited. The SPC paradigm augments the workflow via *Semantic Enrichment*, *KI*, and *KR*—typically injected during *S4–S6*—to enable rule- or learning-based reasoning, automation, and context-aware analysis (Table 1). We map the six stages to the SPC pillars as follows: *S4–S5* correspond to *KE*, with *S1–S3* for preparation; *Semantic Enrichment* and *KI* occur primarily during *S6*; and *KR* is formalized during/after *S6* and connects to the *CDL* for automated reasoning.

As summarized in Table 1, SPC extends the conventional pipeline by incorporating intelligence through KI and KR, facilitating deeper contextual understanding and enabling

adaptive behavior across the asset lifecycle. This baseline positions the review to pursue the objectives by analyzing how SPC implementations realize enrichment, integrate knowledge, and formalize KR across domains in Section 4.

Table 1. Comparative summary of traditional processing vs. SPC-based approaches.

Aspect	Traditional Point Cloud Processing	SPC-Based Approach
Aim and deliverables	Geometric modeling and spatial measurement; outputs are maps/meshes	Semantic modeling oriented to analysis/decision support; outputs are enriched models for intelligent applications
Processing paradigm	Hand-tuned filters and segmentation; per-dataset pipelines	Knowledge- and data-driven pipelines (rules + ML); higher automation and consistency
Semantics and knowledge assets	Limited or implicit semantics; ad hoc metadata and operator expertise	Explicit KI/KR with ontologies, rules/constraints, and links to external knowledge
Reasoning and automation	None or heuristic scripting	Logical/rule-based inference and task automation; integrates with CDL
Lifecycle and integration	Batch-oriented; limited updateability	Incremental updates; links to BIM/DT; lifecycle-aware
Reproducibility and governance	Pipeline-specific; limited Protocolization	Protocolized workflows with explicit schemas/constraints; higher reproducibility

3. Systematic Literature Review (SLR) Methodology

SLRs follow transparent, protocolized procedures to ensure methodological reliability and comprehensive coverage [24]. Guided by the SLR model of Xiao et al. [25] and the Preferred Reporting Items for Systematic Reviews and Meta-Analysis (PRISMA) 2020 guidelines [26], we tailored a six-phase workflow for our review (Table 2). The protocol was not registered/published.

Table 2. Methodological framework for the Systematic Literature Review (SLR) on Smart Point Cloud (SPC), outlining six phases from review question framing to evidence synthesis.

I. PROTOCOL
• Defined objectives to unify SPC terminology and methods. • Formulated six Research Questions (RQs) grounded in initial scoping.
II. SCOPING
• Generated 20 candidate terms; screened for contextual relevance and objectives. • Finalized 3 core search themes: “Smart/Intelligent/Rich Point Cloud”.
III. SEARCH
• Source: Selected 3 search channels (Scopus, Web of Science, and Google Scholar). • Strategy: Defined Boolean queries and 3 inclusion criteria aligned with RQs. • Screening: 4-step filtering (Title >> Abstract >> Full-text >> Recency). • Enhancement: Expanded corpus via Forward/Backward Snowballing.

Table 2. *Cont.*

IV. ANALYSIS
• Extracted 15 elements across Metadata, Scope, Processing, Tools, and Outcomes. • Categorized studies into *Direct*/*Indirect*/*New Use*.
V. SYNTHESIS
• Synthesized evidence narratively, grouped by pipeline stage (S1–S6) and objectives. • Visualized trends via distribution plots and mapping.
VI. INTERPRETATION
• Answered RQ1–RQ5 based on synthesis and proposed modular framework. • Identified open challenges (RQ6) and future directions.

3.1. Review Questions

We formulated six research questions (RQs), grounded in the study objectives and initial scoping of the literature. Table 3 lists the RQs guiding data collection and synthesis.

Table 3. Research questions guiding the SLR on SPC studies.

ID	Research Question
RQ1.	To what extent is the term "Smart Point Cloud" (SPC) used and recognized in the literature?
RQ2.	What alternative terminologies are employed to describe SPC-like concepts across different studies?
RQ3.	How is SPC formally defined, and what are the core elements of its conceptualization?
RQ4.	Why do researchers use varying terms or definitions to describe similar SPC-related concepts?
RQ5.	What data types, methodologies, and processing workflows are used in SPC studies?
RQ6.	What are the key challenges and future research directions related to SPC development?

3.2. Keyword Selection Strategy

Effective keywording determines the relevance, completeness, and manageability of the corpus. An exploratory search using "Smart Point Cloud" [27] yielded 20 candidate terms (Table A1). After screening for relevance to the research objectives, broader algorithmic terms (e.g., 'Point Cloud Processing') and narrow applications (e.g., 'Object Labeling') were excluded for lacking the data-model focus of SPC. We retained three core themes covering 11/20 candidates, providing a focused yet inclusive search scope:

(1) "smart point cloud", (2) "intelligent point cloud", and (3) "rich point cloud".

3.3. Literature Search Strategy

Following PRISMA 2020 [26], we used three identification techniques: *(1) structured database searches*, *(2) backward snowballing* (cited references), and *(3) forward snowballing* (citing references) [25]. Additionally, the authors' domain expertise contributed to identifying key researchers and relevant terms, as recommended in [28]. Searches placed no date limits, and the last search was in June 2025. A PRISMA flow diagram is in Figure 3.

Primary databases included Scopus [29] and Web of Science [30], for indexing reliability. Google Scholar [31] was also included to capture grey literature and broader conference coverage. Query strings (Table A2) were constructed using standard Boolean operators and exact-phrase matching. The search was then conducted in two phases: (1) title-based search across all three databases, and (2) full-text search on Google Scholar using the broader phrase "Smart Point Cloud" (Protocol and results summary in Table A3).

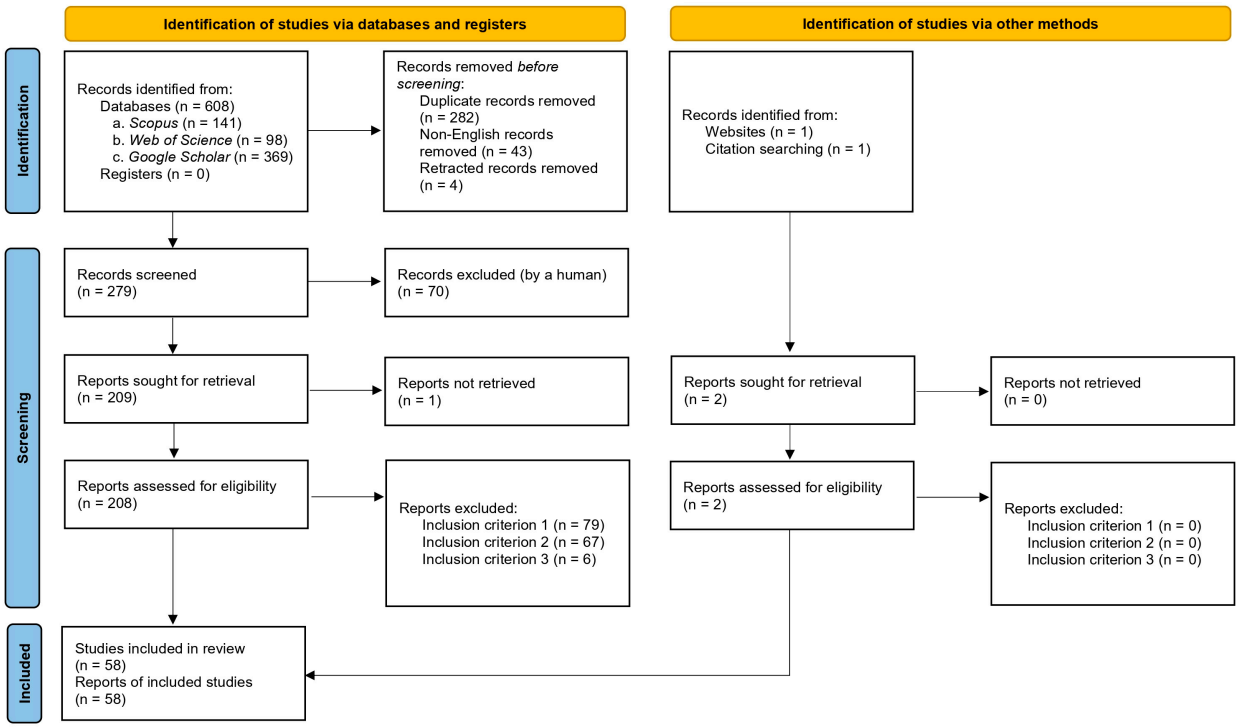


Figure 3. PRISMA 2020 flow diagram for the systematic review summarizing identification, screening, eligibility, and inclusion.

3.4. Screening and Eligibility

3.4.1. Criteria Development

Inclusion/exclusion criteria (Table 4) were designed to reflect the RQs and ensure topical relevance, rigor, and recency. We included peer-reviewed journals, books, book chapters, conferences, theses, preprints, and other grey literature where relevant.

Table 4. Inclusion and exclusion criteria for screening and eligibility.

Type/#	Criterion
Inclusion 1	Addresses SPC's concept and is situated in geomatics, engineering, computer science, or a related field.
Exclusion 1	Falls outside those domains or uses non-point-cloud primary data.
Inclusion 2	Presents novel methods or advanced techniques aligned with SPC objectives outlined in Table 1.
Exclusion 2 [1]	Lacks a clear description of objectives, methods, or results, or shows abstract–full-text inconsistencies.
Inclusion 3	Represents the most recent version/update from the same authors on a similar topic.
Exclusion 3	Superseded by a more recent publication by the same authors on the same topic.

[1] Given the corpus's theoretical frameworks and methodological proposals focus rather than empirical trials, formal risk-of-bias was not applicable. Quality appraisal was integrated into Exclusion Criterion 2, rejecting studies lacking clear objectives, reproducible methods, or consistent results.

3.4.2. Screening Procedure

Screening involved exhaustive manual verification of all retrieved records from the searches without arbitrary cut-offs. First, non-English and retracted items were removed. Duplicates across databases were also eliminated, retaining only the earliest indexed version. Screening then followed four sequential steps:

- Title screening for topical relevance (*Criterion 1*);
- Abstract screening for relevance and methodological alignment (*Criteria 1–2*);
- Full-text screening for alignment with objectives and inclusion criteria;
- Recency screening to keep the most recent version by the same authors (*Criterion 3*).

The core set was then seeded with the authors' domain expertise for relevant but potentially overlooked seminal studies identified during the research framing, after which backward and forward snowballing were applied. All additional candidates underwent the identical screening and eligibility criteria above. Primary screening was performed by the lead author and a random 20% subset independently verified by the co-authors to ensure reliability, resolving discrepancies via consensus.

3.5. Information Extraction and Synthesis

As Chigbu et al. [32] emphasize, effective reviews are guided by a focused analytical lens aligned with their objectives. Guided by the RQs, we extracted specific data items from each included study, categorized into five domains:

- Metadata: (1) year; (2) publisher; (3) study region; (4) keywords.
- Research scope: (5) objectives; (6) SPC definition; (7) application domain.
- Data and Processing: (8) data types and acquisition; (9) point cloud pipeline phases addressed (*S1–S6*, see Section 2); (10) methods used at each phase.
- Implementation: (11) software/tools; (12) evaluation metrics.
- Outcomes: (13) findings; (14) linkage to RQs 1–6; (15) limitations.

The lead author extracted data using a standardized Microsoft Excel form, with 20% cross-verified by co-authors to minimize subjectivity (discrepancies resolved via consensus). Missing items were marked "Not Reported" rather than inferred.

Data were synthesized narratively, grouping studies by SPC contribution, pipeline stage, and RQs. Trends and characteristics were visualized via tabulation and distribution plots. Furthermore, we qualitatively explored methodological heterogeneity by analyzing performance variations across acquisition modalities, processing algorithms, and tools.

4. SLR Results: SPC Definition and Methods

Our systematic search—applying all inclusion and exclusion criteria—yielded a final set of 58 publications, detailed in Table A4. For analytical clarity, we introduce a three-way classification used throughout Section 4:

- *Direct* (n = 22): Studies that explicitly define, propose, or architect a framework (or a specific component thereof) clearly labeled as a "Smart Point Cloud".
- *Indirect* (n = 22): Studies that reference the SPC concept and implement its core capabilities but do not explicitly apply the "Smart Point Cloud" label to their own work.
- *New Use* (n = 14): Studies that utilize the specific phrase "smart point cloud" but refer to unrelated concepts rather than the knowledge-integrated data model defined here.

4.1. Overview and Classification of SPC Literature

The temporal distribution of the categorized references is shown in Figure 4, highlighting how SPC-related research evolved. Earlier related expressions predate SPC—"*Smart Cloud-of-Points*" in 2005 and 2006 within architectural modeling [33,34], and the exact term "*Smart Point Cloud*" in 2010 for city modeling [35]. The SPC concept examined here was explicitly introduced in 2015 by Poux and Billen [36] and developed through a series of *Direct* contributions up to 2021. Subsequent publications are predominantly *Indirect*. This suggests that while core SPC methodologies (e.g., semantic reasoning) are diffusing into practice, recent studies often implement them partially or under alternative labels rather than strict SPC nomenclature, resulting in their '*Indirect*' classification.

Relevant articles often lack standardized keyword entries despite discussing SPC-like ideas. To examine the semantic landscape, we performed a descriptive keyword analysis of *Direct* papers (Figure 5a). High-frequency terms include data/source markers (e.g., "*3D Point Cloud*", "*LiDAR")* and semantic/process terms (e.g., "*Ontology*", "*Semantics*", "*Segmen-*

tation", and "*Classification*"), underscoring SPC's emphasis on semantic enrichment within point cloud pipelines. Application cues (e.g., "*Cultural Heritage*", "*Intelligent Environment*") indicate cross-disciplinary reach.

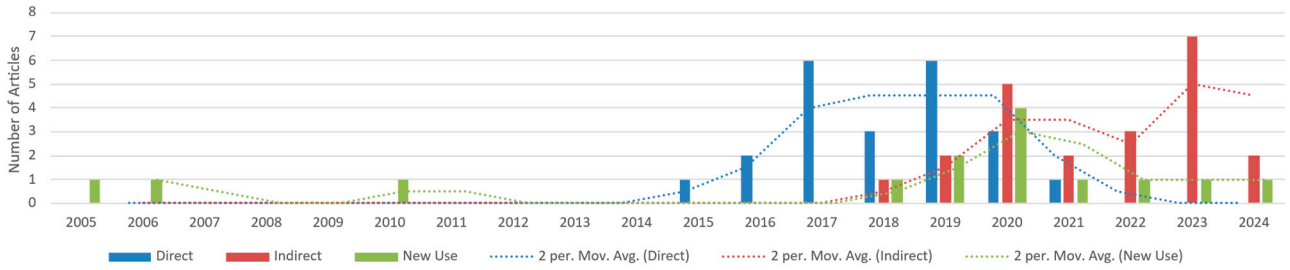
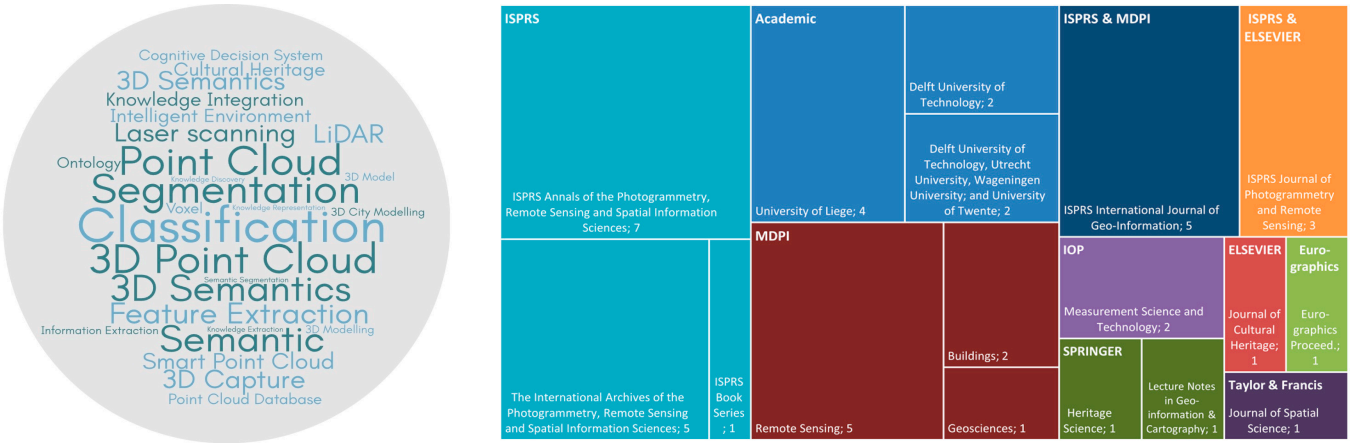


Figure 4. Temporal distribution of *Direct*, *Indirect*, and *New Use* studies (2005–2024).



(**a**) (**b**)

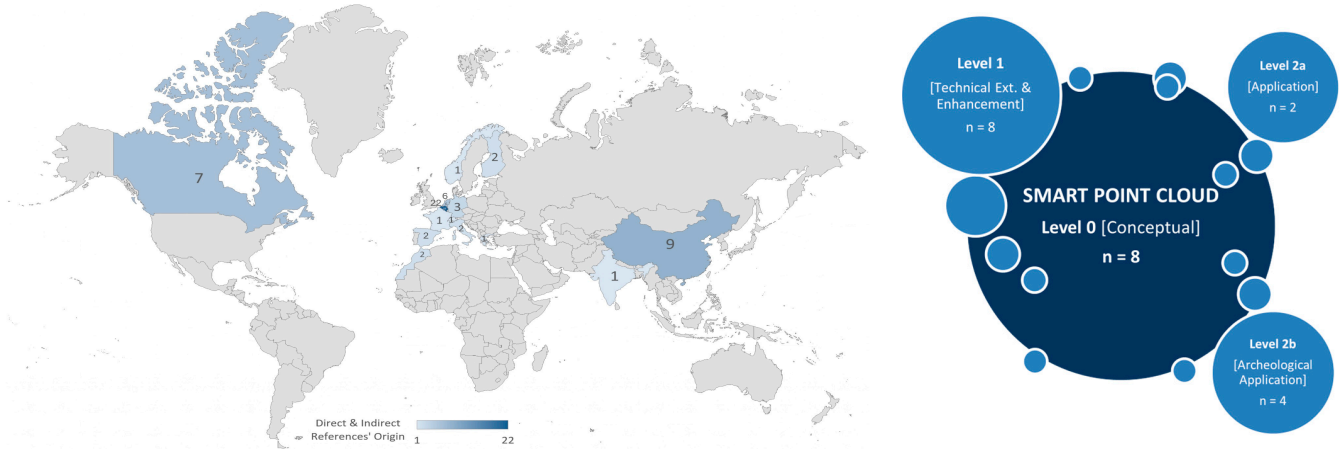
Figure 5. (**a**) Word cloud of high-frequency author keywords in *Direct* studies; font size reflects frequency. (**b**) Treemap of publication venues for *Direct* and *Indirect* references.

Publication venues cluster within geomatics and remote sensing—most notably the *ISPRS Annals*, *ISPRS Archives*, *ISPRS International Journal of Geo-Information*, and *Remote Sensing*—with grey literature outputs from academic outlets (e.g., *University of Liège*; *Delft University of Technology*). A treemap of venues appears in Figure 5b.

Figure 6a maps the geographic distribution of studies. *Europe*—especially *Belgium* (n = 22)—leads SPC publications, reflecting the sustained efforts at the University of Liège, which date back to the concept's origin in 2015 [36]. *China* (n = 9) and *Canada* (n = 7) follow, often through cross-regional collaborations with European institutions. *Counts reflect author-affiliation mentions; a paper may contribute to multiple countries.* This geographic concentration likely drives another terminological drift: as the concept expands beyond its European origin, researchers often adopt the methodology but favor generic terms (e.g., 'semantic point clouds') over the specific 'SPC' label.

Our qualitative assessment of the study aims shows recurring goals—*automating workflows*, *reducing processing time*, and *improving accuracy*. To summarize conceptual maturity, *Direct* contributions were grouped by abstraction level (Figure 6b):

- Level 0: Conceptual/review outlining SPC principles (n = 8; 36.4%) [15,36–42];
- Level 1: Technical extensions of SPC a pipeline stage/s, (n = 8; 36.4%) [43–50];
- Level 2: Application implementations—general (n = 2; 9.1%) [51,52] and archeological (n = 4; 18.2%) [13,14,53,54].



(**a**) (**b**)

Figure 6. (**a**) Global distribution of *Direct* and *Indirect* SPC studies by author-affiliation mentions (non-exclusive counts). (**b**) Distribution of *Direct* studies by conceptual focus level.

4.2. SPC Conceptualization and Definitions

Raw point clouds—though rich in geometry—lack semantics, explicit structure, and connectivity, which limits automated interpretation. As outlined in Section 2, early utilization therefore emphasized spatial structuring (e.g., tiling/octrees), segmentation, and manual or rule-based classification. As pipelines matured, multi-source fusion improved completeness and robustness, and time-varying attributes enabled longitudinal analyses. These advances produced semantically labeled point clouds but did not yet establish the *relational*, *queryable*, *governed* data model needed for automated reasoning. This gap led to the Smart Point Cloud (SPC) concept.

An SPC is a queryable, multi-scale point cloud data model that augments geometry with semantics, topology/relations, provenance and uncertainty, temporal/versioning, and application-specific attributes. Where appropriate, it binds to domain ontologies and exposes interfaces for rules and ML to support automated reasoning, incremental updates, and task-specific analytics. An SPC is a representation, not merely a processing pipeline—although SPC pipelines maintain model consistency as data evolve.

Table 5 summarizes the conceptual evolution from raw point clouds to SPCs. To avoid conflating *representations* with *processing*, the comparison treats four representations: Raw, Structured (spatial organization without full semantics), Semantic (labeled/attributed but no explicit topology or reasoning interface), and SPC (bundles semantics + relations + indices/Levels of Detail (LoDs) + rich queries + governance/automation).

Across both *Direct* and *Indirect* strands, the literature converges on SPC as a knowledge-based data model that fuses spatial, semantic, and topological information to support context-aware analysis and decision support. In this vein, Poux and Billen [15] emphasize integrating geometry and semantics for high-level reasoning, while Poux [39] highlights interactive querying, multi-scale visualization, and system-level intelligence.

Infrastructure-oriented work formalizes SPC as a reasoning-ready platform, specifying how geometry and semantics are integrated and exposed through relations and constraints. These contributions establish the structural preconditions—relations, provenance, governance—on which automation can operate.

According to Poux et al. [40,42], SPC transforms raw data into user-specific information models by combining semantic enrichment with adapted processing techniques (e.g., mathematical morphology and image processing algorithms), improving efficiency (e.g., avoiding full conversions) while maintaining consistency as data evolve. Recent work—such as Poux and Ponciano [44]—shows self-learning automating instance segmentation,

and Poux and Billen [45] frame SPC as a systematic framework unifying semantic segmentation, connectivity relationships, and domain rules to support automated reasoning and classification.

Table 5. Conceptual evolution from raw point clouds to Smart Point Clouds (SPCs).

Dimension	Raw Point Cloud	Structured Point Cloud	Semantic Point Cloud	SPC—Smart Point Cloud
Core geometry (XYZ)	✓	✓	✓	✓
Per-point attributes (e.g., intensity/color)	○	○	✓	✓
Spatial structure/indexing (tiles, octree, LoDs)	–	✓	✓	✓
Derived features (normals, curvature, descriptors)	–	○	✓	✓
Segmentation and classification (semantic labels)	–	○	✓	✓
Topology/relations (adjacency, part-of, constraints)	–	–	–	✓
Queryability	–	Limited (spatial)	Limited (+ attribute)	Rich (+ topological)
Temporal/context (versioning, external sensors/metadata)	–	○	○	✓
Provenance/quality/uncertainty	–	○	○	✓
Automation and feedback (rules/ML; self-improvement)	Low	Low–Medium	Medium	High
Cross-sensor fusion	–	–	○	✓

Legend: ✓ present; ○ sometimes; – absent.

Beyond labeling, Poux et al. [37] see SPC operating as a data-mining/enabling platform: relations and provenance expose structure in otherwise underutilized point cloud data, turning them into actionable insights. Poux and Billen [36] further underscore SPC's advantages for intelligent documentation and geospatial analysis, while noting the need for domain-specific refinement to operationalize the broad conceptual framing.

In sum, SPC is a relation-aware, queryable point cloud data model designed for automated reasoning and lifecycle updates.

Our review also identified 14 "*New Use*" studies, where "smart point cloud" denotes unrelated constructs (e.g., proprietary formats/labels, alternative metrics). These are outside the SPC definition used here but help explain emerging interpretations and terminological drift. Table 6 summarizes these patterns.

Table 6. Summary of "*New Use*" meanings of "smart point cloud" beyond SPC.

Ref.	Label (As Used in Source)	Meaning in Context
[55]	Velodyne Smart Point Cloud	Proprietary software/format for managing point clouds
[56]	"Smart" Point Clouds	Integration of hyperspectral data, classes, and mission metrics with imagery-based point clouds for forestry management.
[57,58]	Smart Point Cloud	Spherical Motion Average Radiant Temperature (SMaRT): thermal fluctuation analysis measure in urban studies.
[59]	Hexagon SPC (HSPC)	Engine/format for storing large point sets with per-point metadata
[3,60–63]	Smart Point Cloud mesh	Close-range photogrammetry (CRP) pipelines labeled "smart" for 3D documentation
[64]	Smart Point Cloud	Structuring methods to enable large-scale point cloud processing
[35]	Smart Point Cloud	High-fidelity point clouds integrated in virtual globes for accessible 3D city modeling
[33,34]	Smart Cloud-of-Points (SmCP)	Parametric/generative control over point-based 3D models

4.3. SPC Workflow Architecture and Methodological Contributions

The transition from conventional point cloud workflows to SPC systems is both incremental and transformative. While foundational operations—acquisition, registration, pre-processing, segmentation, classification, and modeling—remain essential for geometric fidelity and analysis, they do not, by themselves, impart "smartness" through semantical depth, adaptive behavior, or reasoning capacity.

SPC frameworks build on these foundations by integrating automation, semantic enrichment, and knowledge-based inference—particularly during segmentation, classification, and data modeling. This integration adds contextual understanding that enables advanced analytics, adaptivity, and domain-specific decision support, turning point clouds into intelligent, queryable assets.

To conceptualize this transformation, Poux [39] proposed a modular SPC architecture with coordinated server-side and client-side components (Figure 7). On the server-side, the Data Management zone operates within a *Dynamic Processing Layer*, utilizing a parsing module and a processing module (both KE) to partition incoming data and generate functional subsets, which are validated and persisted in the point cloud database module (KI). Above this, the Knowledge Enrichment zone integrates the *Knowledge Layer* (KI/KR) to augment data using ML models, ontologies, and a language/query processor that exposes semantic interfaces. A *reasoner module* implements constraint checking and inference, bridging to the CDL. The *Agent Layer* coordinates autonomous, task-oriented agents for scheduled or event-triggered operations (e.g., re-classification, rule enforcement). Finally, on the client-side (User Interaction zone), a graphical interface supports tasks such as spatial queries, semantic computations, and scenario simulations—enabling collaborative, near-real-time work with semantically enriched 3D data.

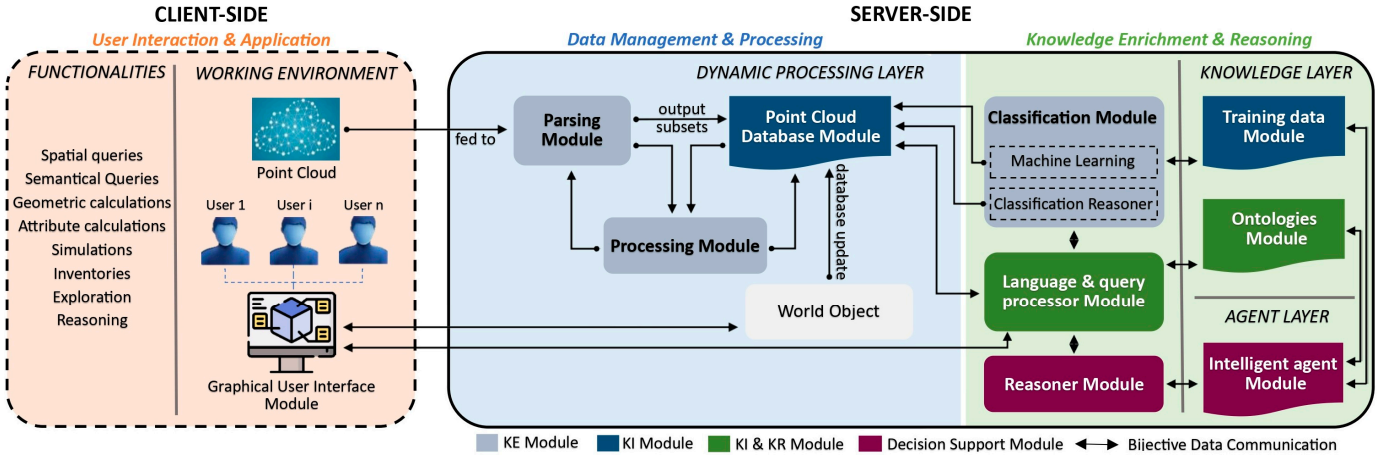


Figure 7. Modular SPC framework visualized in three functional zones: Data Management, handling parsing, processing, and storage; Knowledge Enrichment, where domain ontologies, classification, and agents drive the system's intelligence (both on server-side); and User Interaction, covering client-side visualization and semantic and spatial interaction. Adapted from [39].

This subsection examines how SPC systems contribute at each stage of the broader workflow, emphasizing where and how intelligence is embedded, what data it uses, and which traditional processing is implemented across the capture-to-use lifecycle.

Figure 8 summarizes how *Direct* studies target each stage. Most implementations concentrate on pre-processing (n = 19), segmentation (n = 19), classification (n = 18), and data modeling (n = 17)—the primary loci for embedding semantics, automation, and reasoning. In contrast, acquisition, registration, and 3D modeling are less represented (each $\leq$n = 6), highlighting areas where SPC functionality is limited.

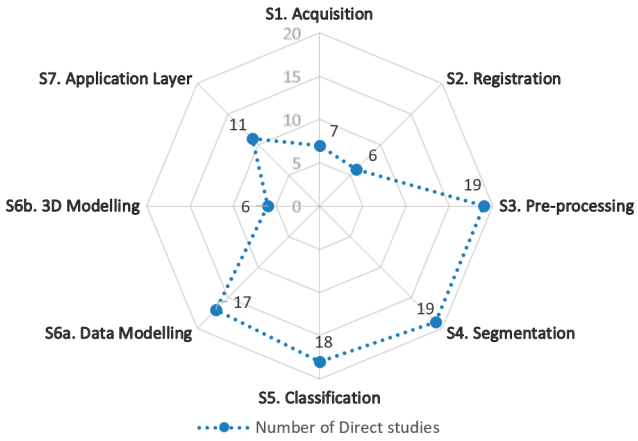


Figure 8. Count of SPC *Direct* studies per each traditional point cloud workflow stage.

4.3.1. Data Collection and Systems

Data acquisition in geomatics uses one or more complementary techniques to obtain fit-for-purpose results. This typically include establishing control networks for geo-referencing and executing site-specific planning and instrument configuration to manage constraints such as limited visibility, complex terrain, or poor satellite coverage [65].

With advancements, contemporary point cloud recording technologies include laser scanning [5], photogrammetry, and RGB-Depth (RGB-D) imaging [66]. These differ in resolution, accuracy, range, cost, and platform flexibility; sensor choice depends on task requirements, downstream algorithms, and resources. Multi-sensor deployments are also becoming common to compensate limitations—e.g., pairing LiDAR with RGB to add color, or combining airborne and terrestrial platforms for completeness. For SPC, acquisition also seeds provenance (sensor model, scan geometry, GNSS/IMU logs) and attribute channels needed later for semantics and uncertainty.

Figure 9 presents visual examples of point cloud datasets acquired using different sensing techniques across various scales, highlighting variation in resolution and detail.

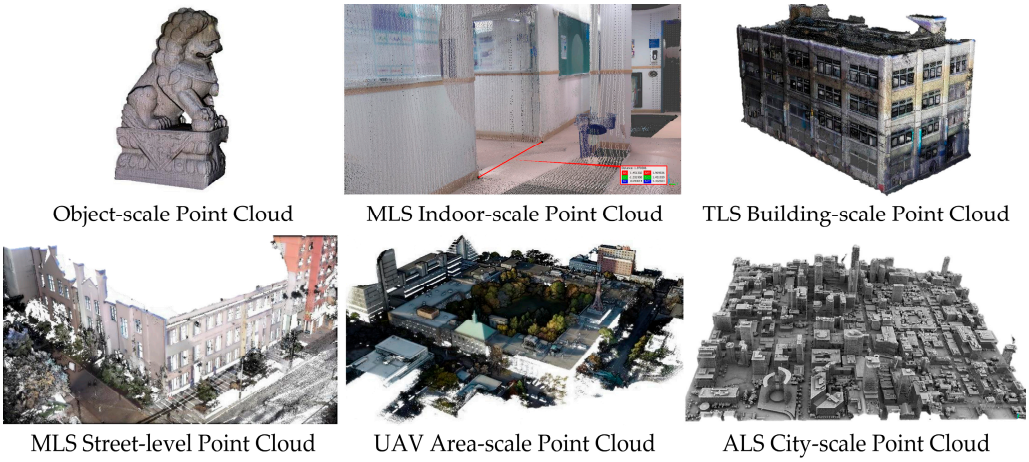


Figure 9. Examples of point cloud data acquired using different techniques across various scales.

To contextualize acquisition strategies within SPC development, Table 7 synthesizes typical collection methods, characteristics, and their frequency of use in the *Direct* studies.

Table 7. Comparative analysis of point cloud acquisition methods and implementation in SPC.

Technology	Type	Characteristics [1]					SPC Implementation		
		Density	Accuracy	Range	Cost	Flexibility	Usage (%)	Application	Reference Studies
Laser Scanning	TLS	VH	VH	MH	H	LM	25.5	Cultural heritage; medium-scale urban settings	[13,14,41,51,53,54]
	MLS–Handheld	MH	M	L	H	VH	16.4	Floor layouts; constrained spaces	[39]
	MLS–Vehicle	M	M	M	H	M	5.5	Large outdoor built environment	[43,48]
	ALS	L	H	VH	VH	MH	7.3	Large urban/landscape coverage	[39,42]
Photogrammetry	CRP	H	LM	L	L	MH	20.0	High visual fidelity	[53,54]
	UAV Photogrammetry	M	LM	MH	M	MH	1.7	Agriculture/forestry	[13,49,52]
	Videogrammetry	LM	LM	LM	LM	LM	0.0	Not used	n/a
Depth Imaging	RGB-D	LM	LM	L	L	MH	0.0	Not used	n/a
Synthetic	Benchmark datasets	VH	VH	n/a	VL	VH	23.6	Framework validation	[44,51]

[1] V = Very; H = High; M = Medium; L = Low; n/a = Not Applicable.

The *Direct* studies utilized 55 datasets across these modalities, often mixing sources within a single experiment. Figure 10 shows the distribution by sensor/platform, underscoring SPC's adaptability to diverse inputs.

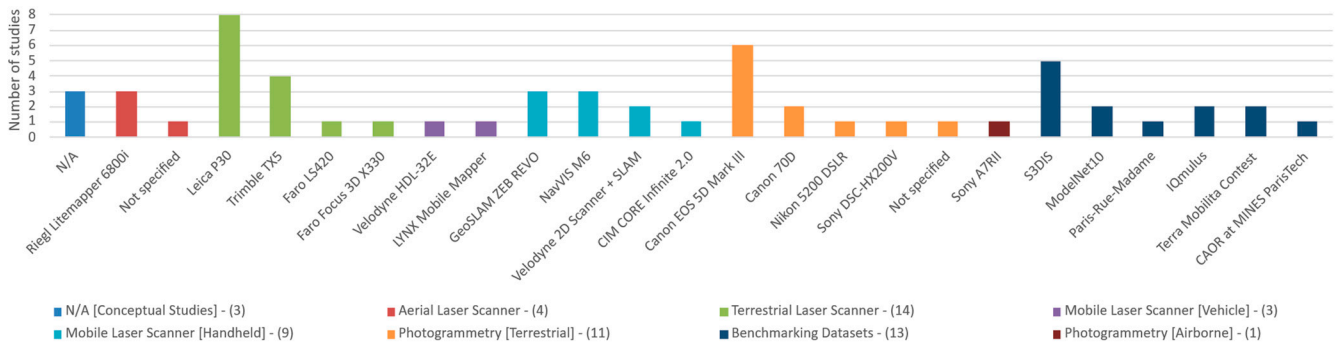


Figure 10. Distribution of sensor/platform types across the datasets used in *Direct* SPC studies (n = 55). Multiple datasets may come from the same study.

While many studies evaluated frameworks on single-source datasets—or multiple datasets independently—(55.6%), multi-source fusion was also common (44.4%), improving resolution, visual richness, and coverage. Fusion of LiDAR and photogrammetry, in particular, can be seen in a few studies, including the following:

- Studies [14,41,53,54] combined Leica P30 TLS for dense geometric precision with Canon EOS 5D Mark III CRP for visual texture in archeology;
- Yang et al. [13] employed CIM CORE Infinite 2.0 handheld MLS, Faro LS420 TLS, and Sony A7RII UAV photogrammetry to document cultural relics across scales;
- Poux et al. [42] integrated Riegl Litemapper 6800i ALS, Trimble TX5 TLS, and image-based Sony DSC-HX200V data for indoor/outdoor heritage building;
- Van Bochove [47] merged ALS for rooftop and topographic mapping, complemented by MLS for façades and street-level detail.

In conclusion, SPC frameworks readily integrate heterogeneous acquisition sources. Strategic sensor selection—guided by precision, cost, and contextual relevance—directly influences data richness and downstream processing, while multi-source fusion aligns with SPC's goal of building comprehensive, provenance-aware models and with the complexity of real-world environments.

4.3.2. Data Alignment and Integration (Registration, Geo-Referencing, and Fusion)

Registration estimates the transformations that align multiple geospatial datasets—acquired from different positions, platforms, or sensors—into a common spatial frame [67] (Figure 11). When alignment is tied to a geodetic reference (e.g., World Geodetic System 1984 (WGS 84)), it is *Geo-referencing*, which may occur during acquisition via GNSS, per-scan with targets, or post hoc using Ground Control Points (GCPs). As noted in Section 2, registration and pre-processing may interleave depending on sensor and workflow.

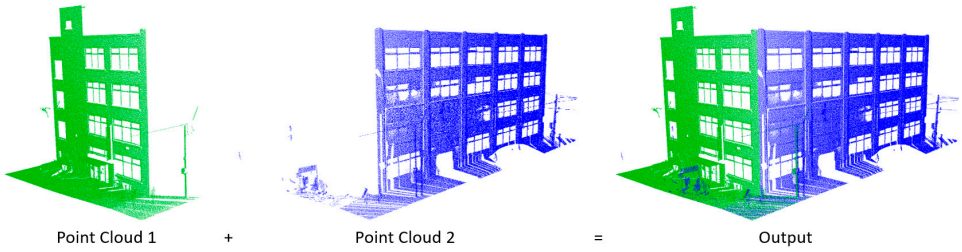


Figure 11. Example of pairwise registration: two scans (green/blue) aligned into a common frame to produce the fused output.

Meanwhile, *Fusion* integrates multiple spatial datasets and/or attributes to increase completeness and semantic richness, and can take place in two forms:

1. Geometric Fusion (register-then-merge): co-registered datasets of the same format (e.g., point clouds) from different modalities (e.g., ALS + TLS + CRP) are merged to improve coverage and consistency while preserving the native characteristics of each dataset; optional post-fusion duplicate removal or density weighting may be required to reconcile overlaps and sampling differences to produce a coherent output.
2. Attribute Fusion (register-then-transfer): heterogeneous sources (maps/images/point clouds) are first co-registered so that units (e.g., pixels, points) correspond one-to-one, enabling per-unit attribute propagation (e.g., colorizing or texturing LiDAR from orthophotos or transferring normals/material cues) to enrich the target dataset.

This subsection focuses on alignment methods used in SPC workflows to prepare data for robust fusion and downstream semantics.

Registration in point clouds typically proceeds in two phases: a coarse step for approximate pose estimate, followed by a fine step that refines alignment to point-level accuracy. Robust registration underpins applications such as autonomous navigation [68], posture estimation [69], and 3D reconstruction [70].

For coarse registration, historical practice relied on manual point pairing or target-based methods. Recent work introduced learning-based approaches that (i) extract features and establish correspondences [71] or (ii) predict transformations end-to-end [72]. These methods are promising at scale and in dynamic environments but remain sensitive to sampling density, scale differences, and the breadth of training data [73].

For fine registration, classical ICP [74] remains dominant, iteratively minimizing distances between corresponding points to compute the optimal transformation per user-defined iterations and thresholds. Variants such as Generalized ICP (GICP) [75] and Normal Distributions Transform (NDT) [76] further improve robustness with noisy or variable

density data. However, these methods often rely on a good initialization and may struggle with complex or repetitive structures [77].

Across the *Direct* SPC corpus, registration is essential for aligning datasets and enabling fusion—where most value accrues—and was found in a supporting infrastructure rather than the primary locus of innovation. Most studies follow typical pipelines: control- or feature-based coarse alignment followed by ICP-family refinement, sometimes with knowledge-guided priors. For example, Poux et al. [42,53] used GCPs and targets for initial geo-referencing, then ICP; Van Bochove [47] similarly adopted ALS as a global reference, co-registering MLS data via GCPs and ICP in succession. Feature-based registration explored by Poux [39], aligning geometric primitives (planes, corners) for coarse lock prior to ICP. Poux et al. [54] further integrated non-uniform rational B-splines (NURBS) surface fitting coupled with ICP to improve accuracy through knowledge-based geometry. Meanwhile, Joosten [48] applied template matching and pose graph optimization (PGO) using 2D map priors, GNSS, and IMU for large-scale alignment. A comparative summary of the *Direct* SPC study methods is provided in Table 8.

While not a core of innovation in SPC, it should be noted that accurate alignment ensures that relations (adjacency/part-of/connectivity) constructed later on are geometrically valid, and allows provenance and uncertainty to be propagated consistently into the SPC model, supporting KI/KR consistency and reliable reasoning.

Table 8. Registration methods reported in *Direct* SPC studies—suitability, initialization, limitations, and computational cost.

Stage	Method	Suitability	Initialization Needs	Limitations	Comp. Cost
Coarse	GCPs/ targets	Stabilizing scenes; alignment to global reference frame	Surveyed GCPs; network geometry	Driven by control quality and network geometry	Low–Moderate
	Feature-based	Scenes with clear planes/ edges/corners for initial lock	Feature extraction; good spatial distribution	Degrades with weak/ambiguous features; clutter	Moderate
Coarse	Template match/PGO	City/corridor-scale alignment with map priors (GNSS/IMU)	Map archives; GNSS/IMU; match constraints	Requires reliable multi-modal priors; graph tuning	Moderate
Fine	ICP	Local refinement of overlapping scans	Reasonable initial pose; overlap; outlier filtering	Sensitive to poor initialization or repetitive geometry	Moderate–High
	ICP + NURBS	Engineered/architectural surfaces; parametric surface fit	Pre-fit NURBS surfaces in proximity	Pre-fit required; high compute for surface fitting	High

4.3.3. Point Cloud Preparation (Pre-Processing)

Pre-processing prepares raw point clouds for downstream analysis by stabilizing geometry, harmonizing attributes, and reducing volume without discarding information. These are particularly critical for multi-source fusion workflows: because platforms differ in sampling density and radiometric response—and real scenes contain noise, outliers, and gaps—rigorous normalization is required to make datasets comparable across sources and sessions [78]. For SPC, pre-processing also records provenance (parameters, software) and seeds uncertainty that can be propagated into semantics and reasoning. This subsection summarizes key pre-processing operations reported in the *Direct* SPC studies (Figure 12) and how they integrate into multi-step flows.

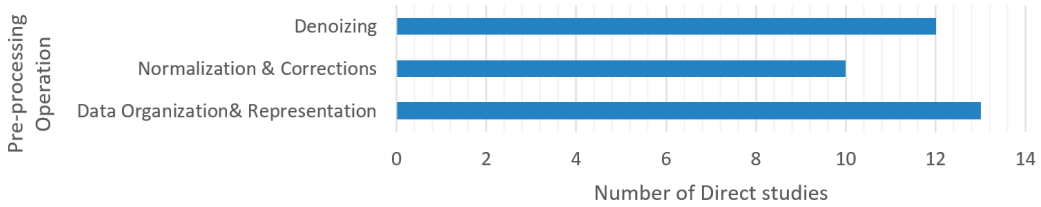


Figure 12. Distribution of key pre-processing operations reported in the *Direct* SPC studies.

Noise and Artifact Removal—Filtering in SPC is typically the first step, removing spurious points while preserving scene structure (Figure 13). Parameterized filters are common; for example, Kotoula [49] applies Local Outlier Factor (LOF) to improve subsequent Principal Component Analysis (PCA)-based normal estimation and surface-orientation reliability. In SPC pipelines, denoising often precedes alignment/fusion to improve cross-sensor consistency—Poux [39] highlights statistical outlier removal before integrating TLS, MLS, and photogrammetry; Van Bochove [47] removes transient artifacts (vehicles, pedestrians) to reduce overlap conflicts and redundancies.

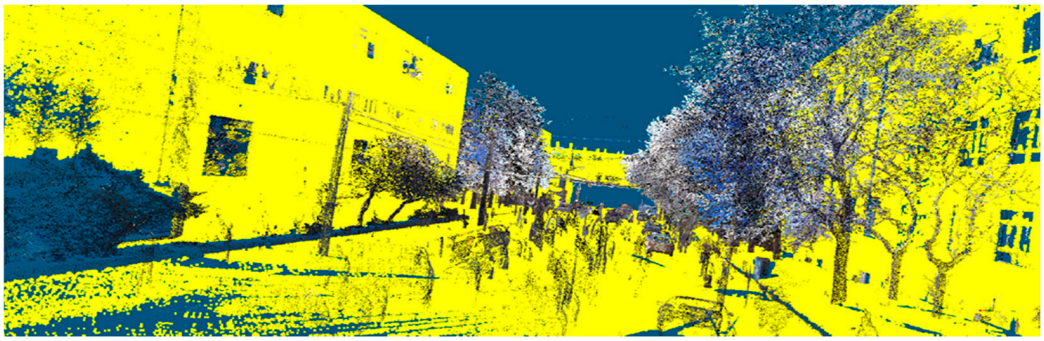


Figure 13. Example of denoising through non-ground/building removal in an urban scene.

Normalization and Corrections—Normalization imposes uniform characteristics to improve feature comparability and fusion. Common steps in SPC include density equalization, range-based bias corrections, intensity/radiometric calibration, and normal-orientation disambiguation. Balado et al. [43] emphasize aligning object shape, size, orientation, and density with benchmarks. SPC multi-sensor workflows commonly follow denoising → normalization/correction → alignment → fusion. For example, Poux et al. [14] filter intensity overloads and mixed pixels, apply intensity corrections and geometric alignment, then fuse TLS + photogrammetry; Poux et al. [53] combine denoising, density adjustment, geometric alignment, and full-waveform analysis prior to fusion.

Density control and Organization—Subsampling reduces computational load while preserving morphology and supports hierarchical storage (Figure 14). Approaches in SPC include Middle of Octree (MidOc) subsampling used by Zheng [46] for progressive geometric approximation, followed by denoising, normalization, and redundancy pruning to enable efficient ordering and Convolutional Neural Network (CNN) ingestion. At a larger scale, Joosten [48] tiled clouds into 50 × 50 m grids to simplify management before filtering and to derive 2D density histograms for template matching and map alignment. Furthermore, octrees/voxels in SPC serve double duty: (i) as LoD indices for multi-scale visualization, queries, and storage, and (ii) as substrates for superpoint/region graphs that support KI/KR aggregation and constraint reasoning (see Section 4.3.4).



Figure 14. Octree-based subsampling for density regularization and hierarchical storage.

Takeaways for SPC pre-processing can be summarized as below:

- Denoising is the most frequent and typically first step; along with normalization, both facilitate fusion to ensure cross-sensor geometric/attribute comparability.
- Data organization materially improves storage/runtime and ML ingestion and provides the indices that underpin SPC LoDs and region graphs.
- SPC pre-processing pipelines are composite and sensor-aware; filters and normalizations are tuned to modality and the downstream task.
- Provenance capture (filter types/parameters, residuals) should be written into the SPC's metadata to support reproducibility and uncertainty-aware reasoning.

4.3.4. Feature Extraction and Hierarchical (Region) Segmentation

Feature extraction computes descriptors that make classes separable and robust to sensor noise, density variation, and viewpoint changes by quantifying local shape, appearance, and context for each point or supervoxel [53]. These descriptors fall into three categories: geometric (normals, curvature, planar/cylindrical fits) capturing shape and orientation [79]; radiometric (RGB/Hue-Saturation-Intensity (HSI) statistics, LiDAR intensity), useful where material/reflectance contrasts exist [80]; and contextual (relative height, neighborhood statistics, adjacency/graph cues, spatial topology) that disambiguate semantically similar shapes by setting [81].

A typical four-step pipeline is used: (1) neighborhood definition per point/supervoxel via fixed-radius (scale-aware) or k-nearest neighbors (k-NN; sampling-aware) search; (2) primitive estimation, fitting local planes and/or running PCA on the neighborhood covariance to obtain normals, curvature, and eigenvalue-based shape indices; (3) descriptor computation aggregating geometric, radiometric, and contextual cues; and, optionally, (4) multi-scale aggregation by repeating across multiple supports and/or octree levels, or by propagating descriptors to supervoxels/segments to balance fine detail with structural context. The neighborhood model thus strongly determines normal/curvature stability and—ultimately—segmentation quality [53].

Segmentation partitions points/supervoxels into meaningful components that support downstream labeling, interpretation, and analysis; it uses the descriptors above to decide which points belong together [23]. Common families include the following: model/primitive fitting (e.g., Hough Transform (HT) [82], random sample consensus (RANSAC) [83]) to fit planes/cylinders and assign inliers—effective for dominant parametric structures but computationally heavier and less robust in heavy clutter or deformation; region growing/connectivity clustering, expanding from seeds whose neighbors exhibit similar features—sensitive to thresholds and neighborhood design [84]; graph-based partitioning, constructing a k-NN or voxel-adjacency graph, weighting edges by feature similarity and spatial smoothness, then optimizing a cut/merge objective (e.g.,

spectral or normalized-cut variants); and density-based clustering (e.g., density-based spatial clustering of applications with noise (DBSCAN), mean shift), which handles noise and non-parametric shapes but is sensitive to scale under variable density.

SPC frameworks emphasize segmentation that captures semantic geometries, relations, and feature similarity while leveraging device-specific, domain-specific, and expert knowledge [37]. In our *Direct* corpus, 54.5% (12/22) studies [14,15,38–41,44,45,51–54] adopt a three-stage hierarchical segmentation design:

1. Voxel/Octree Partitioning—All 12 studies above use hierarchical voxel subdivision (octrees) to structure the scene into grid-based indices for efficient storage, retrieval, and feature computation; recursion is controlled by minimum voxel size, maximum depth, points-per-voxel, or application rules (e.g., color thresholds).

Poux et al. [40] introduced a conceptual multi-scale voxel structure supporting diverse data resolutions coupled with preliminary filtering and, later, semantic attributes. Poux et al. [54] extend this to fused LiDAR + photogrammetry with nested octrees to optimize rendering and organization. This stage structures and over-segments the data but does not yet decide semantic groupings.

2. Feature Extraction—Within the voxel framework, studies employed multiple techniques to compute task-specific cues, for example:

ML-driven graph clustering over voxel neighborhoods, linking adjacent voxels whose descriptor similarity exceeds learned thresholds, consolidating neighborhoods prior to region grouping [39]; PCA-based analysis to derive principal directions and plane normals, often followed by voxel-to-voxel contact ("touch") topology to assess planarity relations (horizontal, vertical, mixed, neighboring) [39,44,51]; shape descriptors and plane fitting (notably RANSAC) for geometric extraction [14,39,41,52–54]; elevation-based separation with Cloth Simulation Filter (CSF) to isolate ground/non-ground [52]; and radiometric cues (RGB/HIS, intensity) to enhance material differentiation—particularly in archeological settings where shape alone is impractical [14,39,41,53,54].

3. Grouping via Connected Component Labeling (CCL) on a region-adjacency graph (RAG)—The voxel/supervoxel lattice yields a RAG with standard three-dimensional connectivities (6/18/26). Edges are weighted by descriptor similarity (e.g., normals/curvature, radiometry), spatial smoothness, and primitive/topology constraints (collinearity, orthogonality, planar contact). 3D CCL—originally a 2D image-analysis method—then groups nodes into connected elements ("semantic patches"), region units that are spatially and semantically indexed.

Putting these stages together, Poux and Billen [15] describe the SPC mechanism that first organizes points into a voxel space at an octree level set by device-specific parameters. Each voxel is analyzed for features and local similarity to define "*Connected Elements*"—derived from initial voxelization and feature extraction of primary components (e.g., ground, perpendicular/parallel elements) that act as potential parents. Hybrid voxels (containing multiple potential objects) are further subdivided using surrounding topology/characteristics. The process yields "*Semantic Patches*", categorized as pure voxels or leaf nodes. Table 9 summarizes the three stages, inputs, operations, and outputs.

Beyond this pattern, some studies utilize alternative pipelines: Kotoula [49] combines image-based boundary detection (Canny, Hough) on panoramic imagery with photogrammetric point clouds derived from the same panoramas; 2D segments are re-projected onto the point cloud and refined with 3D HT to enhance edge delineation. Zheng et al. [50] use fixed-radius vertical cylinders as per-point neighborhoods to compute vertical profiles (density/height statistics), facilitating feature extraction. For simpler requirements,

Joosten [48] groups points using map-extracted building footprints and buffers, while Yang et al. [13] rely on manual annotation for cultural relics.

Table 9. Three-stage hierarchical region segmentation used in the *Direct* SPC studies.

Stage	Purpose and Inputs	Core Operations (Examples)	Outputs
1. Voxel/Octree Partitioning	Structure/over-segment scene; input: raw points	Build octree; stop by voxel size/max depth/ points-per-voxel; maintain adjacency	Cells/blocks (multi-resolution index), adjacency
2. Feature Computation	Compute cues per point/supervoxel using local supports	Neighborhoods; PCA (normals); primitive fits (planes); radiometry (RGB/HSI); contextual/topological measures; multi-scale aggregation	Descriptors at point/supervoxel/octree levels
3. Grouping via CCL on RAG	Group elements using similarity + adjacency + constraints	Build RAG; edge weights = feature similarity + spatial smoothness + primitive/topology constraints; 3D CCL (6/18/26 connectivity)	Connected elements/"semantic patches" (regions)

Eventually, the resulting semantic patches/regions become the units consumed by knowledge-based classification and reasoning in Section 4.3.5, where domain rules and KI/KR formalize labels and support automation.

4.3.5. Knowledge-Based Classification

We distinguish semantic segmentation (one semantic label per point/voxel) from classification (one label per entity—segment/object or scene). Instance segmentation augments semantic segmentation with instance IDs. Here, we focus on segment/object classification of the regions produced by Section 4.3.4.

The *classification* stage assigns semantic labels to segments/objects, enabling targeted analyses (e.g., vegetation, buildings, roadways), change detection, and direct use in Geographical Information Systems (GIS)/BIM workflows where typed entities are required [78]. In SPC pipelines, classification benefits from both data-driven learning and knowledge-based constraints derived from device, domain, and expert priors. Manual classification, however, appears in some SPC studies (e.g., [13,47,52]) and can achieve high precision, but it is labor-intensive and scales poorly; these works often advocate automating labeling for complex objects and larger datasets [47].

Beyond manual labeling, learning-based classification includes classical ML models such as Support Vector Machine (SVM) and Random Forests (RF), trained on handcrafted geometric/radiometric/contextual attributes; they scale to large datasets but require careful class-imbalance handling and intensity calibration [85].

Deep learning (DL), on the other hand—point-based (e.g., PointNet/PointNet++), voxel-based (e.g., sparse 3D CNNs), and projection-based (e.g., range/multi-view) architectures—learn features end-to-end but require balanced training data and explicit handling of density variation and domain shift [86]. For example, PointNet/PointNet++ operate on unordered sets with permutation invariance and alignment modules, supporting global (segment/scene) classification [87], while Dynamic Graph CNN (DGCNN) replaces PointNet's multi-layer perceptron (MLP) aggregation with EdgeConv layers over dynamic k-NN graphs in feature space, treating point clouds as structured graph data composed of nodes and edges to capture local geometric context at multiple layers, although at the cost of model complexity and training parameters [88]. Voxel/projection CNNs leverage spatial hierarchies, learn intricate patterns, and achieve strong accuracy—widely documented in automated classification—but incur quantization/projection trade-offs and potential infor-

mation loss [89]. In Smart Point Cloud studies, deep backbones are typically configured with global heads to produce segment-level predictions by pooling features across semantic patches (or by applying graph neural networks over the RAG), while noting that the same architectures can be configured with per-point heads for semantic segmentation.

Knowledge-based Classification (rules and morphology)—Rule-based systems use expert rules, geometric/topological constraints, and graph reasoning to assign object types efficiently where domain knowledge can systematically guide decisions. Attributes such as size, shape, orientation, connectivity, and adjacency are encoded as thresholds/rules at the segment/object level to classify entities such as walls, ceilings, and furniture [14,41,45]. SPC studies emphasize semantic enrichment via spatial feature attributes as a core mechanism for structuring point clouds [42]. For instance, studies [43,48,49] explored morphological and rule-based detectors for specific classes, deploying morphological opening/closing and geometric heuristics to isolate patterns such as steps (stairs), slender verticals (poles), or tubular forms (pipelines)—achieving high precision in constrained scenes but limited generality. In parallel, Zheng et al. [50] and Poux et al. [37] highlighted the effectiveness of classical ML (SVM/RF), or CNNs, as in [46], training on attributes such as reflectance, elevation variance, and point density; improving scalability while remaining compatible with domain knowledge.

Ontology-driven Classification—Ontology-driven approaches formalize domain knowledge in structured machine-interpretable ontologies (classes, relationships, and constraints)—using standards such as Web Ontology Language (OWL)—to support logical reasoning and consistency checking; these are widely used in SPC [15,39,40,44,51,53,54], often with SPARQL Protocol and Resource Description Framework (RDF) Query Language (SPARQL) for querying. This extends knowledge-based methods by enabling schema-aware inference and traceable decisions, with reported gains in tasks such as material property analysis and geometric feature extraction. Coupled with a self-learning mechanism, SPC classification proceeds in two steps:

1. Rule Definition: Expert-prescribed semantic rules define each class and are tailored to data and sensor characteristics; variations in expected and actual representations (e.g., geometric discrepancies due to occlusion) may reduce accuracy;
2. Rule Refinement: To address such inaccuracy, rules are dynamically adjusted from the data itself in this step—adjusting the initial rules by analyzing characteristics shared among similar classified segments—to make criteria more robust. For example, computing confidence intervals over numeric region properties (sizes, heights) for segments of the same class [37].

Hybrid (neuro-symbolic) Classification—Combining ML/DL predictions with ontology/rule constraints can further improve consistency, especially on long-tail classes. A proposed pattern is as follows: (i) ML to propose labels (with confidence) for semantic patches; (ii) ontology/rules to validate/correct labels using RAG relations (e.g., doors on walls, floors under walls) and cardinality/compatibility constraints; (iii) conflicts are to be resolved via policy (e.g., rule-overrides, confidence thresholds); and (iv) provenance and confidence are written to metadata for auditability and downstream reasoning.

Table 10 compares the main classification families in the *Direct* SPC studies by suitability, accuracy, inputs, and computational cost.

In conclusion, SPC classification typically operates on semantic patches/regions from Section 4.3.4, with graph-based reasoning over a RAG to enforce relational constraints. Neuro-symbolic hybrids can use ontological constraints to validate/correct ML predictions, improving robustness on rare classes. Classification outputs should persist labels, confidences, rule traces, and conflicts in SPC metadata to support auditability and further reasoning. Performance metrics are discussed in Section 4.6.

Table 10. Comparative characteristics of classification approaches in *Direct* SPC studies.

Method	Suitability	Accuracy [1]	Inputs/Data	Compute Cost [1]
Manual/ Interactive	Small datasets; gold-standard labeling; expert QA	Very high; operator/inter-annotator dependent	Segmented data + annotation tools	High (labor-intensive)
Classical ML (SVM, RF)	Small–medium labeled sets; interpretable features;	High with good features; sensitive to class imbalance	Handcrafted features (normals, intensity)	Moderate–High
Deep Learning (global heads)	Automation at scale; complex patterns	High with sufficient data; drops under domain shift	Raw points/voxels + labels; model params	High (training)
Knowledge-based (rules)	Rule-driven environments with strong priors	High when rules hold; brittle out-of-domain	Expert rules + Attributes/metadata	Low–Moderate
Morphological and Rules	Targeted geometric elements (poles, stairs, pipelines)	High on targeted classes; limited flexibility	Morphological features; thresholds	Low–Moderate
Ontology-based (OWL + rules)	Complex semantics; consistency; explainability	Very high on schema-covered classes; explainable	Ontology + rules; attributes; queries	High (reasoning)
Ontology + Rule Refinement	Adaptive/changing environments	Improves with validated updates; schema-bound	Initial rules + feedback-driven statistics	High (iterative)
Hybrid (ML + ontology/rules)	Production systems needing accuracy + constraints	Better consistency than ML alone; resolves contradictions	ML predictions + ontology constraints	Moderate–High

[1] Accuracy and computational cost are qualitative and depend on data scale and class balance.

4.3.6. Three-Dimensional Models vs. SPC Data Models

The *modeling* stage turns processed point clouds into an application-ready representation via: (1) a geometry-centric path that reconstructs explicit or implicit surfaces for editing, simulation, and exchange; or (2) a data-centric path that retains points as the primary geometry and adds semantics, relations, and indices for reasoning and analytics. Modeling thus determines *what* is represented (surfaces vs. enriched points), *how* it is organized (hierarchies/LoDs), and *how* it interoperates with external systems. This subsection corresponds to S6: Data Modeling in Section 2 and consumes the segmented regions from Section 4.3.4 and the labels from Section 4.3.5.

In the geometry-centric path, explicit 3D models store surfaces directly as polygonal meshes or parametric primitives (e.g., Bézier, NURBS), while implicit models define surfaces as level sets of a volumetric function (e.g., Poisson, signed-distance fields (SDFs)), from which meshes are extracted as needed. Although explicit reconstructions are compact and highly editable for Computer-Aided Design (CAD)/BIM, inferring stable control points/primitives from noisy or incomplete scans is challenging, whereas implicit reconstructions yield watertight, smooth surfaces but may over-regularize fine details [51,90]. SPC data models may combine both ideas while retaining points as the source of truth. Table 11 contrasts the two paths.

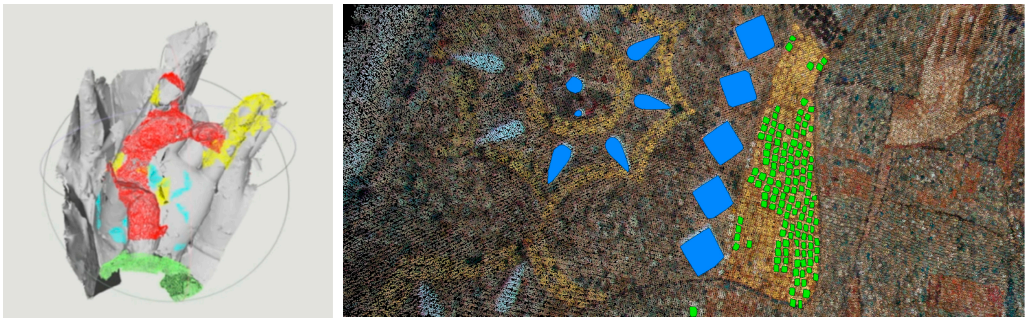
1. Three-dimensional Models (geometry-centric)—Surface-based models remain essential in some SPCs where downstream tools or standards require surfaces/solids. For example, Yang et al. [13] apply 3D modeling to support mesh-based analysis and virtual restoration within an SPC framework (Figure 15a); Van Bochove [47] generates high-resolution façade and roadway polygon meshes from multi-source point clouds; and Poux et al. [51] use context-aware procedural modeling (informed by ontolo-

gies and shape matching) to integrate voxel-organized point clouds with parametric assemblies across multiple LoDs, improving recognition and reconstruction.

2. SPC Data Model (data-centric)—Most SPC studies avoid full mesh generation, focusing instead on semantic enrichment and structured point clouds. With semantic patching—organizing segmented points into attribute-bearing units—introduced by Poux et al. [42], and the extension via hierarchical aggregation to form objects and relations as suggested by Poux and Billen [15,45], studies like Kharroubi et al. [52] and Zheng [46] validated direct, out-of-core rendering and interaction over these enriched point clouds; further work (e.g., Poux et al. [37,40,54]) demonstrated reasoning/querying over SPC data in practical applications (Figure 15b).

Table 11. Geometry-centric 3D modeling vs. data-centric SPC data models.

Dimension	Three-Dimensional Modeling (Geometry-Centric)	SPC Data Model (Data-Centric)
Representation	Explicit meshes or parametric surfaces (Bézier/NURBS); or implicit fields (SDF/Poisson) with mesh extraction	Points remain the source of truth; regions/supervoxels; attributes + relations (adjacentTo, part-of, constraints)
Semantics and reasoning	Usually retrofitted (labels in BIM/CityGML or sidecars); limited native reasoning	Built-in semantics with ontology links; queryable; constraint checking and reasoning
Update model	Remeshing/refitting can be expensive; parameter-sensitive	Incremental enrichment; append/patch attributes; provenance and uncertainty can be carried forward
Scale and performance	Dense meshes are heavy; watertightness may over-smooth	Octree/tiling LoDs; out-of-core storage and multi-LoD queries
Interoperability	Strong with CAD/BIM/CityGML and simulation stacks	Strong with database (DB)/graph stacks (SQL/SPARQL); export meshes only when needed
Best-fit use cases	Editing/design, fabrication, simulation, standards requiring surfaces	Analytics/monitoring, search, lifecycle ops, uncertainty-aware decision support



(**a**) (**b**)

Figure 15. (**a**) A geometry-centric 3D mesh with downstream damage analysis. Adapted from [13]. (**b**) Query with results executed directly on enriched points in SPC framework. Adapted from [53].

Hierarchical Tiers: SPC data models commonly adopt multi-tier layers that align with Sections 4.3.4 and 4.3.5 [46,48]: classifying data into essential strata and advancing to contextual interrelations and domain-specific ontology integrations at elevated tiers. Poux and Billen [15,45] delineate point cloud representation with Level 0 (L0) = *semantic patches* (Primary elements: over-segmented, attribute-bearing regions), Level 1 (L1) = *connected elements* (Secondary elements: aggregates constrained by topology/relations), and Level

2 (L2) = *domain objects* (Transitional elements: ontology instances with roles/relations). These tiers encode explicit relations (e.g., part-of, adjacentTo, constraints) and support element-level queries and visualization (e.g., roof vs. façade layers) [37,40,53].

Data Management and Querying: Operational SPCs require a three-layer design: (i) *physical organization*, (ii) *logical schema*, and (iii) *access/query*. The physical layer uses octree/tiling as an LoD/storage index for out-of-core access (e.g., Potree-style layouts used in [13,52]). The logical schema stores attributes and provenance per L0/L1/L2 entity and formalizes relations (e.g., part-of, adjacentTo, constraints) in an ontology/typed graph; examples include Unified Modeling Language (UML)-based schemas as used by Kotoula [49], and voxel/patch-centric Point Cloud Database Management System (PC-DBMS) designs in Poux et al. [51,54]. The access/query layer pairs Structured Query Language (SQL) (tabular attributes/joins) with SPARQL (schema-aware queries and reasoning); hybrid models such as Multiple Interpretation Data Model (MIDM) proposed by Billen et al. [41] co-locate both, enabling multi-view access and inference. These designs scale storage and query while preserving the semantics, relations, provenance, and uncertainty that SPC requires.

To summarize, a 3D model path is needed when downstream delivery mandates surfaces/solids (CAD/BIM/CityGML, simulation, fabrication), while an SPC data model is preferable for rapid updates, large-scale analytics, uncertainty retention, or schema-aware search/reasoning. We suggest that systems can combine both: maintain an SPC for enrichment and analytics, then export targeted surfaces or parametric assemblies only where the delivery requires them.

4.4. SPC Applications and Use Cases

Point cloud data powers many domains (mapping, Architecture, Engineering, and Construction (AEC)/Facilities Management (FM), heritage, environment). Here, we focus on what SPC adds in practice—semantics, queryability, reasoning, lifecycle updates, and uncertainty retention—drawing on evidence from the review corpus. Table 12 summarizes SPC's value by domain; corpus-grounded vignettes follow.

Table 12. SPC applications summarized by domain (evidence from *Direct*/*Indirect* studies).

Domain	Typical Data	SPC Capability	Outputs	Refs.
Urban planning and Development	ALS/MLS/ TLS + imagery	Zoning/height/adjacency queries; LoS/ sun analyses filtered by semantic classes; rule checks against city schemas	Compliance flags, scenario analytics, impact assessments	[15,47]
Cultural Heritage	TLS/CRP /UAV	Multi-scale semantic patches → domain objects; temporal/condition layers; provenance and uncertainty for traceability	Queryable digital archives; targeted restoration and monitoring	[13,15,39]
AEC/FM	TLS/MLS + BIM	Asset typing; scan-to-BIM consistency; deviation analysis and maintenance queries over typed elements	Issue lists, as-built vs. design diffs, work orders	[47,49,51]
Transport and Roads	MLS/ALS	Street-asset typing; road-surface classes; network/logical constraints for QA	Inventories, safety checks, maintenance maps	[46,48]
Environment and Natural Resources	ALS/UAV (+ waveform)	Knowledge-based ground/non-ground; canopy/strata labels; temporal stacking with uncertainty	DTMs/CHMs, change/volume analysis, risk layers	[39]

Urban planning and Development—SPCs in urban planning move beyond geometry-only layers by encoding classes (building, roof, road) and relations (adjacentTo, within) together with height rules, enabling planners to run volumetric analysis such as zoning, height, and adjacency checks as well as line-of-sight (LoS) and solar exposures scoped to semantic classes rather than raw points. Practical implementations, as in Poux and Billen [15] and Van Bochove [47], combine multi-source data for comprehensive coverage, preserve provenance of sources and parameters, and can pass rule-checked results into GIS for scenario testing and impact assessment—yielding compliance flags and annotated alternatives for planning decisions. Visualization products can further communicate these semantically filtered scenarios to non-expert audiences, making complex spatial relationships accessible for public and stakeholders' engagement.

Cultural Heritage—SPC pipelines in heritage transform point clouds into multi-scale semantic patches aggregated into domain objects with condition and temporal attributes as well as uncertainty, producing queryable, provenance-aware archives for targeted restoration, and longitudinal monitoring, as in [13,15]. Interactive systems such as Poux [39] further demonstrate real-time navigation and semantic queries (e.g., by material or condition) with multi-LoD inspection, allowing curators to track changes across campaigns and to document decisions with explainable, rules-checked outputs.

AEC and FM—Within AEC and Facilities Management, SPC models type assets such as walls, slabs, and mechanical, electrical, and plumbing (MEP) components, and then drives deviation analysis and scan-to-BIM consistency checks directly over semantically filtered regions. Image/photogrammetry-enriched point clouds combined with SPC rules (e.g., cylinder fitting with domain thresholds as implemented by Kotoula [49]) support reliable pipe and utility localization and attribute queries for maintenance and safety workflows in manufacturing settings. Geometry-centric deliverables (with SPC retained as the source of truth for enrichment, analytics, and updates) are also demonstrated by Poux et al. [51], applying reverse engineering for furniture modeling to support analysis and prototyping. Van Bochove [47] further integrated SPCs with BIM systems in indoor environments, supporting issue lists and work orders grounded in explainable criteria to improve maintenance scheduling, resource management, and lifecycle monitoring of building components. These efforts can be further explored in construction monitoring through automated as-built versus design diffs to identify discrepancies in project elements and timelines, facilitating early corrections and improving project accuracy.

Transport and Roads—Corridor-scale SPCs produce street-asset inventories in which signs, poles, guardrails, and related fixtures are typed and logically anchored to carriageways with placement and clearance rules, enabling automated quality assurance (QA) for completeness and violations [46]. The same semantics can also support road-surface classification and visibility/proximity checks for sightlines and obstacles inside GIS, yielding prioritized maintenance maps and safety interventions traceable to the SPC labels and constraints used to generate them [48].

Environment and Natural Resources—In environment and natural resources, knowledge-based SPC rules improve ground/non-ground separation and canopy stratification to derive digital terrain models (DTMs)/canopy height model (CHM), while temporal stacking with uncertainty supports change and volume analysis for erosion, stockpiles, and vegetation. Additionally, as highlighted in [39], full-waveform attributes further enrich SPC features with material cues that sharpen classification and monitoring fidelity, resulting in risk layers and analytical products that carry explicit confidence metadata for conservation and resource management decisions.

4.5. Implementation Tools Analysis

SPC studies rely on a mix of photogrammetry suites, point cloud processing libraries, databases/ontology stacks, and rendering frameworks. This diversity improves fit-for-purpose deployments but can hinder portability. Here, we synthesize the tools reported across the studies and how they are used within SPC workflows, offering a sensible default stack for future implementations. Figure 16 summarizes adoption by category; Table 13 consolidates representative libraries and stacks.

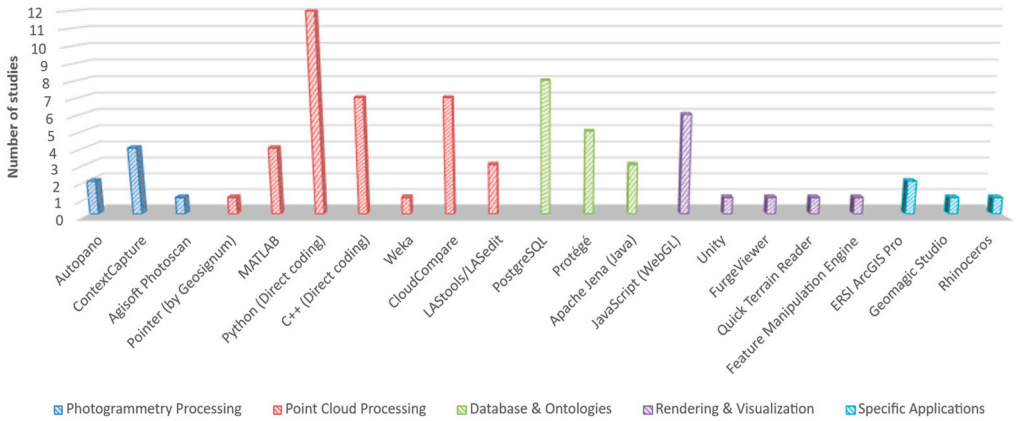


Figure 16. Software used in the *Direct* SPC studies, grouped by area of use. Bars show the number of studies reporting each tool.

Table 13. Use and associated libraries/languages of software identified in the *Direct* SPC Studies.

Software	Use and Associated Libraries/Add-Ons/Languages
	Photogrammetry Processing
Autopano	Stitching panoramic images.
ContextCapture/ Agisoft Metashape	Image-based point cloud reconstruction.
	Point Cloud Processing
GeoSignum Pointer	Cloud ingestion, QA, registration, and feature detection.
Python (Coding)	Image processing (Pillow), ETL (NumPy, laspy, tensorflow, scikit-learn), graph/connectivity inference (math; networkx), visuals (matplotlib), DB link (psycopg/psycopg2, JSON), RDF connection (rdfLib), template matching (SPARQLWrapper; OpenCV), and graphical user interface (GUI) (Django, psycopg2, json).
C++ (Coding)	Classification (PDAL, GDAL, PCL; LIBSVM); and visuals (CCLib).
MATLAB (Coding)	Prototyping, and classical ML classification.
CloudCompare	Registration, segmentation, and classification.
LAStools/edit/FME	ETL (Including thinning, and manipulation).
	Database and Ontologies
Protégé/Apache Jena (Java)	Author ontology (OWL/OWL2) over RDF graph model, and connects with SPARQL query language + Pellet reasoner.
PostgreSQL	PC-DBMS; (+ PostGIS and PgPointcloud).
	Rendering and Visualization
FurgeViewer/QTR	Local inspection and visualization.
FME	Inspection overlays.
Unity	VR visualization.
JavaScript (WebGL)	Index-visuals-interacts (Potree; Three.js; jQuery; tween.js, Plas.io).
	Specific Applications
ESRI ArcGIS Pro	Map edits/visualization.
Geomagic Studio	Mesh deformation analysis.
Rhinoceros	NURBS/mesh creation.

4.5.1. Photogrammetry Processing

Studies producing image-based point clouds most frequently used Agisoft Metashape (formerly PhotoScan) and ContextCapture for dense reconstruction and orthophoto generation [14,42,53]; with Autopano for panoramic stitching where relevant [49]. Choices were typically justified by reconstruction robustness, control over camera models, and batch automation.

4.5.2. Point Cloud Processing

GeoSignum Pointer (reported only in [47]) is a government cloud platform that covers ingestion, pre-processing, inspection, registration testing, and targeted feature detection for ALS/MLS datasets. Its end-to-end hosting simplifies workflows, though reliance on the provider's cloud can constrain access and portability.

For general coding stacks, Python was the orchestration language in ~55% of papers (e.g., Refs. [45,51,54]) given its rich ecosystem (e.g., NumPy, and Open3D) for voxelization, segmentation, classification, and data modeling. However, its limited execution speed in real-time implementations prompted some studies such as [39,40] to hand off hot spots to C++ for compute-intensive kernels (~32%), using PDAL, GDAL, and PCL, keeping Python for pipeline control. MATLAB (language and software) also appears in ~18% (e.g., Refs. [43,46]), mainly for prototyping and classical classifiers (e.g., k-NN), sometimes alongside Python for numerical/image operations [15,49].

Among desktop tools, CloudCompare is the most common workbench (31.8%; e.g., Refs. [39,47,52]) for interactive registration, segmentation trials, inspection, and visualization—its open access and broad feature set make it a versatile academic option. For pre-processing, LAStools/LASedit found useful for thinning/manipulation (13.6%; e.g., Refs. [46,47]), and Feature Manipulation Engine (FME) used for format/attribute extract, transform, and load (ETL) into downstream systems.

4.5.3. Database and Ontologies

Ontology tooling centers on Protégé software for ontology authoring and descriptions (22.7%, e.g., Ref. [15]) using OWL/OWL2 over the RDF graph model, with built-in SPARQL querying language support and pluggable description-logic reasoner engines (commonly Pellet). Similarly, Apache Jena—a Java framework—also appears in some studies (13.6%, e.g., Ref. [51]) to author OWL vocabularies, interconnect ontologies, and employ the RDF model and SPARQL queries and rule-based inference.

For geospatial storage and querying, PostgreSQL is the sole DBMS reported (36.4%; [37,40,41,47,49,54]), typically extended with PostGIS (for geospatial data) and PgPointCloud (for point storage and management). Spatial indexing relies on Generalized Search Tree (GIST) to accelerate window/nearest operations in retrieval [53]; PgPointCloud's tiling/blocking aligns with the octree/LoD organization described earlier.

With the help of Python libraries, typical integration glue consists of psycopg/psycopg2 for DB access and RDFLib or SPARQLWrapper for ontology operations; unique values link table rows (e.g., semantic patches, connected elements) to ontology instances, allowing metadata captured in SQL to be surfaced to the knowledge layer. While effective, reported PostgreSQL limitations include index build/maintenance times and input/output (I/O) for very large clouds, plus the need for substantial database/spatial expertise [49]. At scale, we suggest that teams keep numerically heavy operations in SQL while materializing schema-specific subsets to RDF for SPARQL/OWL checks—preserving performance without sacrificing explainability.

4.5.4. Rendering and Visualization

In addition to the widely used CloudCompare for local visualization and analysis, software like FurgeViewer [46] offered straightforward access to attributes, and Quick Terrain Reader (QTR) provided robust functionality for comprehensive examinations. Additionally, FME [47] facilitated inspection overlay—joining point attributes with external inspection results/test data and exporting harmonized products to analysis or web stacks. For immersive use, Unity has been used to stage SPCs in VR [52], for selection/measurement on streamed LoD tiles and semantic filtering in-scene; useful for stakeholders' engagement.

Web-based implementations commonly rely on JavaScript (27.3%) through Web Graphic Library (WebGL) stacks [39]—Potree and Three.js [53]. Libraries such as jQuery, tween.js, and Plas.io were also used to enhance results. While Potree provides hierarchical LoD streaming (octree tiles), eye-dome lighting (EDL), point budgets, and attribute-based styling/filters that map well to SPC affordances (e.g., filter by class/confidence/uncertainty; select at L0/L1/L2), Three.js was used—e.g., Refs. [47,51,54]—for its powerful custom viewers/overlays (semantic highlighting, interaction tools), improving the interpretability and communication of complex spatial information, although it can face performance challenges with dense point clouds. Optimization through server-side tiling/decimation, multi-LoD streaming, point-budget limits, and client-side shader splats/deferred shading are often necessary to maintain interactivity.

In SPC contexts, renderers typically expose SQL/SPARQL-backed queries—e.g., "doors on walls", "low-confidence" regions, provenance/temporal layers—so visualization acts as an interface to KI/KR rather than a passive viewer.

4.6. Key Performance Metrics and Parameters

This subsection synthesizes how point density, accuracy/precision, and processing time were reported across the corpus, emphasizing trade-offs that matter for SPC pipeline design. Because values across studies are not strictly comparable without aligned datasets, splits, and hardware, we foreground trends and orders of magnitude. Where relevant, we tie observed effects to SPC levers described earlier.

4.6.1. Point Density Considerations

Data density—points per unit area, or its reciprocal, point spacing—shapes what can be reliably detected and how stable features are. Image-based pipelines and laser scanning produce different sampling densities and radiometry, and hence accuracy and robustness. Image-based methods can be rapid and cost-effective for small, detailed targets and yield textured clouds suited to visual analysis, but are less efficient over large extents; laser scanning scales better and delivers higher, more uniform densities, with heavier storage/I/O costs. Hybrid capture that fuses modalities is often recommended to balance strengths. Kotoula [49], however, showed that increasing image overlap, resolution, and count improves point spacing (e.g., ~3 cm spacing $\approx$ ~1100 points/m^2 on planar floors) while achieving ~1 cm registration root mean square error (RMSE) in an indoor SPC application—underscoring the leverage in acquisition parameters.

Furthermore, heterogeneous density and sparse regions degrade classification and inflate the variance of local descriptors; Poux et al. [51] observed that inconsistent densities across combined datasets materially impacted feature extraction and downstream labeling, with dense urban scenes outperforming sparse rural ones, often requiring higher point density. In cultural heritage contexts, Poux et al. [53] reported >95% classification accuracy in areas of high texture quality and low blur, while Billen et al. [41] emphasized the trade-off: higher densities aid recognition but raise processing and storage costs, so density should be matched to the target LoD and task.

SPC-specific mitigations: Pipelines in Sections 4.3.3 and 4.3.4 address density effects via density-aware neighborhoods (radius scaling), uniformization (e.g., MidOc subsampling) to produce stable multi-LoDs, and normalization/corrections prior to fusion—steps that stabilize features and improve comparability across sensors.

4.6.2. Accuracy and Precision

We group metrics by task, semantic/instance labeling commonly reports intersection-over-union (IoU), F1, and overall accuracy (OA), while geometric tasks (registration/reconstruction) report RMSE against references.

For learning-augmented SPC, Poux and Ponciano [44] report an unsupervised segmentation plus ontology-based categorization with self-learning that forms semantic clusters in a voxel framework; they report very high OA on planar-dominant classes of the S3-DIS dataset (99.99%), but lower effectiveness for structural elements, with non-weighted IoU of 49.9 across 13 classes (vs. 42.2 in [45])—reported among the top tiers of benchmarked DL methods. In indoor "as-built" contexts, Poux et al. [51] noted RMSE of 2–5 cm on non-planar elements relative to reference surfaces, with parametric primitive fits under-fitting small deviations; thus, they suggested adding local triangulation/Poisson meshing to improve geometric fidelity. Complementing deep models, Poux [39] showed that supervised pipelines (e.g., PointNet variants with added training data) and domain-specific decision trees can achieve competitive F1 (>85%) on planar-dominant classes. In an SPC heritage application for tesserae recognition, per-class accuracies reached 95% (gold), 97% (faience), 94% (silver), and 91% (colored glass).

SPC-specific levers: Beyond feature design (Section 4.3.4), knowledge-based and ontology-driven constraints (Section 4.3.5)—e.g., rules such as "door adjacentTo wall" or "floor under wall"—help correct or validate ML predictions, improving consistency on long-tail classes without requiring ever-larger training sets. On the geometric side, multi-LoD modeling and queryable provenance/uncertainty (Section 4.3.6) reduce over-regularization and make residuals auditable.

4.6.3. Processing Time Analysis

Runtime reflects scene complexity, cloud size, data layout (tiling/LoD), and implementation/hardware (language, threading, central and graphical processing units CPU/GPU). Poux et al. [51] highlight that processing duration scales with shape complexity, reporting single-threaded classification timings of ~85 s (ModelNet10 dataset), ~32 s (TLS dataset), and ~16 s (S3-DIS) on indoor environments and furniture. Optimized algorithms and efficient data structures can overcome such delays; Poux and Billen [45] reported an automated segmentation throughput of ~1.5 million points/minute, and noted that voxelization is memory-intensive but parallel-friendly. Furthermore, in SPCs step-wise costs vary: in [51], "data mining" to rank and retrieve candidate shapes took up to ~30 s (accelerable via pre-indexing), Constructive Solid Geometry (CSG) operations < 5 s, and the end-to-end pipeline (SPC extraction, multi-LoD modeling, form matching) ~5 min/scene.

SPC-specific levers: The physical organization of data (octree/tiling LoDs, out-of-core access) and selective queries (Section 4.3.6) bound working-set size; Python-for-orchestration plus C++ kernels (PDAL/PCL) and GPU-backed learning stacks (Section 4.5) reduce wall time for heavy stages; and hybrid neuro-symbolic flows (Section 4.3.5) offload parts of the decision process to fast, explainable rule engines, trading expensive retraining for constraint checks.

In conclusion, the synthesis of reported metrics indicates distinct cross-study trends:

1. Density Trade-offs: Higher and more uniform density improves feature stability and classification accuracy, particularly in complex heritage or urban scenes, though this incurs proportional increases in storage and I/O overhead [41,49,51,53].
2. Hybrid Consistency: Accuracy benefits significantly from pairing learned descriptors with SPC's relational/ontological checks, which improve consistency on long-tail classes compared to pure learning approaches [39,44,45].
3. Scalability: Runtime scales efficiently when SPC data models exploit LoD indexing and out-of-core access, with hot paths offloaded to compiled kernels or GPUs [46,52].

4.7. Outcomes

Across the *Direct* papers, several report measurable gains over baseline pipelines when SPC enrichment (semantics, relations, ontology checks) is applied—e.g., higher F1/OA/IoU in labeling tasks or lower RMSE in geometric evaluations (see Section 4.6 for task-specific metrics and examples) [39,43]. In parallel, deployable web viewers were demonstrated in application settings—archeology [53] and indoor mapping [51]—exposing semantic queries and multi-LoD navigation directly on SPCs, which signals movement from prototype workflows toward operational portals.

Given the heterogeneity of the 58-paper corpus and our six research questions (RQs), contributions are uneven (Figure 17).

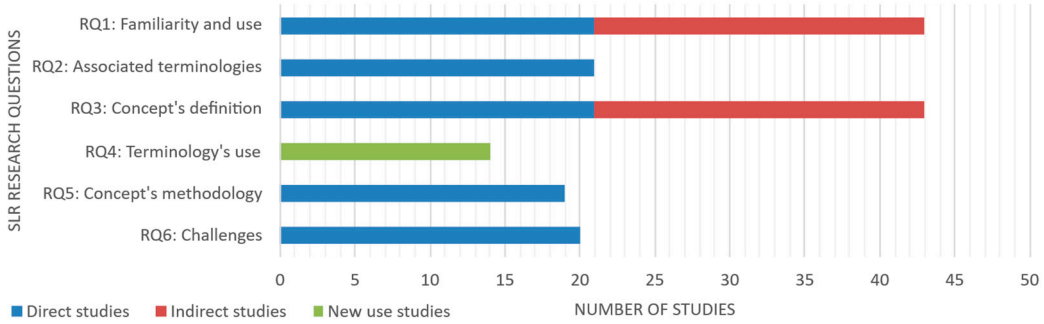


Figure 17. Studies contributing to each RQ. Bars show the number of papers per category.

Considering the full set of 58 studies, 74.1% (n = 43/58) contributed to Q1 (familiarity) and Q3 (definitions). In contrast, Q2 (terminology), Q5 (methodology), and Q6 (challenges) are supported primarily by *Direct* studies: 36.2% (21/58), 32.7% (19/58), and 34.5% (20/58), respectively. Finally, the entire *New Use* set (n = 14/58; 24.1%) chiefly informs Q4, documenting alternative meanings/usages of "smart point cloud".

5. Discussion

This section synthesizes what the corpus establishes about Smart Point Clouds (SPCs) and turns those findings into a modular, implementation-ready view of the pipeline. We first answer the research questions (RQ1–RQ5), then distill a practical framework. Challenges and future work (RQ6) follow in Section 6.

RQ1—Familiarity and Use: Publication quantities and timing indicate that, although increasingly acknowledged, the SPC concept is still consolidating relative to established point cloud representations: *Direct* SPC studies are few in number (n = 22, latest in 2021), alongside recent growth in *Indirect* studies (n = 22) concurrent with broader automation and digitization initiatives. Usage spans multiple domains. In cultural heritage, SPCs serve as queryable, provenance-aware archives supporting conservation, restoration, and monitoring. In urban contexts, SPCs encode classes and relations to run analyses scoped to semantics rather than raw geometry. In AEC/FM, SPCs type assets and drive scan-to-BIM workflows from semantically enriched regions. This breadth shows SPC is adaptable but

must be tailored to domain requirements—heritage prioritizes micro-LoD, provenance, and uncertainty tracking, whereas city-scale planning emphasizes coverage, rule checks, and scenario evaluation. Overall, the corpus supports SPC as a recognized, cross-domain representation; the central implication is how semantics, LoDs, and governance are configured per domain, which sets up the terminology considerations in RQ2 and the methodological choices in RQ5.

RQ2—Terminology and Search Implications: Alternate labels to "Smart Point Cloud" did not perform equally well for retrieval or conceptual fit. During scoping and formal searches (Tables A1 and A2), "Intelligent Point Cloud" returned no records that satisfied our inclusion criteria—likely reflecting its prevalent use in vehicle/robotics sub-literatures rather than geomatics-centric SPC work—while "Rich Point Cloud" yielded only two title-level hits [35,40]. By contrast, "Smart Point Cloud" appeared in 42/44 *Direct/Indirect* included publications (95.5%) when counted across titles/abstracts/full text, aligning most closely with the SPC themes of semantic enrichment, knowledge/ontology use, and enhanced utilization of point clouds. We also observed frequent co-occurring indexing terms (e.g., ontology/ontologies, semantic information, semantic segmentation), which are not synonyms for SPC but proved useful query expanders in practice. Finally, 14 "*New Use*" papers were found employing "smart point cloud" for unrelated constructs (e.g., proprietary formats or thermal metrics); these were retained only to analyze usage drift (see Section 4.2) but excluded from SPC synthesis.

Overall, while 'Smart Point Cloud' remains the definitive anchor for this paradigm, future reviews capturing implicit implementations might expand queries to the functional ontology/semantics synonyms identified, and using the operative checklist defined in RQ3 as a filter, while screening for *New Use* senses to maintain conceptual precision.

RQ3—What is an SPC? Definition and Scope: The literature converges on the need for a precise SPC definition, mainly due to varied terminology use across classification-centric and integration-centric strands such as in [46,91], which complicates synthesis and interoperability. The most frequently referenced conceptual base was the one by Poux (2015) [36], extended by subsequent works (e.g., Refs. [15,38]) and broadly aligned with the S1–S6 workflow analyzed in Section 4.3. Building on that lineage and the corpus evidence, we adopt the following operative definition:

> ***Smart Point Cloud*** *is a* ***queryable***, ***multi-scale point cloud data model*** *that augments geometry with* ***semantics***, ***explicit relations/topology***, *and* ***provenance and uncertainty***, *is* ***bound (where appropriate) to a domain ontology***, *and supports* ***rule-/ML-driven automation and incremental updates*** *across the data lifecycle.*

To make this testable, an SPC (as used in this review) must provide:

- Semantics at point/region/object levels (typed classes and attributes);
- Relations/Topology (e.g., part-of, adjacentTo, constraints) over regions/objects;
- Queryability (rich, selective access over geometry + semantics + relations);
- Provenance and Uncertainty captured and traceable at ingest and during processing;
- Multi-LoD/Hierarchy (e.g., octree/tiling, semantic patches → objects);
- Ontology Hooks (OWL/RDF or equivalent schema for reasoning/consistency); and
- Updateability and Governance—optional (incremental enrichment without full rebuilds; auditable changes).

This definition distinguishes representation from pipeline: SPC pipelines (segmentation, classification, KI/KR) maintain these properties as data evolves; they are not the definition itself. It also clarifies boundaries: a raw or merely labeled point cloud without explicit relations, query interfaces, and provenance is not an SPC; likewise, a mesh-only reconstruction without schema/relations belongs to the geometry-centric path (Section 4.3.6).

In practice, SPCs reduce routine conversions to surfaces for analytics and monitoring, while still exporting meshes/solids when downstream tasks require them.

Finally, standardizing on this feature set provides a common yardstick for comparing methods (RQ5) and for diagnosing gaps and challenges (Section 6), while acknowledging that implementation details—and ontology depth—will vary by domain.

RQ4—Why do multiple, conflicting "Smart Point Cloud" meanings exist, and what are the effects? Our search surfaced 14 papers that use "Smart Point Cloud" for eight distinct, often unrelated meanings (e.g., a proprietary software/format label; a thermal metric acronym; mesh/3D-model pipelines; parametric/organic modeling constructs). Early usage (e.g., 2010 [35]) is not connected to later SPC frameworks, and several post-2019 papers also adopt domain-specific reinterpretations. This polysemy appears driven by disciplinary silos and evolving toolchains rather than a shared conceptual core.

This variability has practical consequences: it injects noise into literature retrieval and screening, obscures comparability across studies, and risks misattribution when "SPC" is cited without definition. In this review, we mitigated the issue by classifying records into *Direct*/*Indirect*/*New Use* and applying explicit inclusion/exclusion criteria (Section 3), so that the synthesis of SPC methods (RQ5) rests on conceptually aligned sources.

To reduce future ambiguity, we recommend a lightweight standardization step: authors who use "SPC" should (i) include a short operational definition aligned with the properties adopted in this paper (see RQ3); (ii) add controlled keywords (e.g., ontology-linked semantics, region topology, multi-LoD, queryable provenance); and (iii) declare whether they mean an SPC data model or a geometry-centric model exported from it. Community venues (special issues, workshops) can further codify the minimal reporting standards proposed in Section 6 to improve cross-domain reuse.

RQ5—What methodologies are used to implement SPC, and how are pipelines composed? Across the corpus, SPC is method-agnostic yet patterned: studies assemble pipelines from established point cloud operations, adding semantic enrichment and rule-/ontology-based reasoning where it creates the most value (Section 4.3). Flexibility is evident in sensor choice (single vs. multi-sensor), learning depth (handcrafted vs. deep features), and Knowledge Integration (rules/ontologies), allowing SPC to accommodate both minimal XYZ clouds and richly attributed datasets. Practically, most *Direct* papers concentrate methodological innovation in pre-processing, segmentation, classification, and SPC data modeling, with acquisition, registration, and explicit 3D surfacing addressed more sparingly (see Section 4.3, Figure 8). Hybrid pipelines mixing geometric cues with ML/DL are common (e.g., Refs. [14,15,40,45,54]), and multi-modal applications (e.g., Ref. [13]) demonstrate that SPC can merge diverse methods and sources without a single mandated stack. The unifying theme is not the specific algorithm but how outputs are made queryable, auditable (provenance/uncertainty), and constrained by domain rules.

Proposed Modular SPC Processing Framework

Grounded in those patterns, we outline a modular, configurable framework. Each module reflects choices from the corpus, documented in Sections 4.3–4.5 (See Table 14); as a software-agnostic architecture, tools listed are illustrative examples, while selections should be tuned to data scale, accuracy targets, budget, and evolving software stacks.

1. Acquisition and Fusion—Choosing sensor mixes that balance coverage, resolution, and radiometry (e.g., TLS/ALS/MLS + imagery), and planning for later fusion by capturing control/metadata (GCPs/trajectories).
2. Pre-processing and Registration—Denoising → normalization → coarse alignment (control/features) → fine alignment (ICP/GICP/NDT) → merging; then structuring into octree/tiles for LoD and out-of-core access. See Sections 4.3.2 and 4.3.3.

3. Hierarchical Segmentation—Building a voxel/octree index → computing geometric/radiometric/contextual features → grouping via CCL on a RAG. Typical stacks use Python orchestration with libraries such as *NumPy* or *Open3D*; see Section 4.3.4.
4. Knowledge-based Classification—Applying expert rules and graph/topology constraints (size/orientation/adjacency) to label segments/objects; coupled with ontology-driven reasoning (OWL/SPARQL) and self-learning rule refinement, where shown effective. See Section 4.3.5.
5. Structuration (SPC data model)—Organizing outputs into tiers (e.g., L0 semantic patches; L1 connected elements; L2 domain ontology instances) with explicit relations (adjacentTo, constraints) and queryable provenance/uncertainty. See Section 4.3.6.
6. Storage and Indexing—Persisting points/attributes in a spatial DBMS (e.g., PostgreSQL + PostGIS + PgPointCloud) aligned with octree indexing; see Section 4.5.3.
7. Access and Reasoning—Using SQL for tabular/selection joins and SPARQL/OWL for schema-aware queries and consistency checks; integration often uses middleware connectors (e.g., psycopg) and semantic libraries (e.g., RDFLib, SPARQLWrapper) coupled with standard reasoners (e.g., Pellet); see Section 4.5.3.
8. Visualization and Interaction—Local viewer/analysis in lightweight software (e.g., CloudCompare); web delivery via 3D rendering frameworks (e.g., JavaScript's Potree or Three.js) for interactive visualization with LoD and attribute filters, supported by backend servers (e.g., Python's Django, psycopg2, and JSON); see Section 4.5.4.

Table 14. Evidence Map linking the proposed pipeline modules to supporting *Direct* studies.

No.	Pipeline Module	Supporting Studies
1.	Acquisition and Fusion	[13,14,41,42,47,53,54]
2.	Pre-processing and Registration	[14,39,42,43,46–49,53,54]
3.	Hierarchical Segmentation	[14,15,38–41,44,45,51–54]
4.	Knowledge-based Classification	[14,15,37,39–46,48–51,53,54]
5.	Structuration (SPC Data Model)	[13,15,37,40–42,45–49,51–54]
6.	Storage and Indexing	[37,40,41,47,49,54]
7.	Access and Reasoning	[15,39,40,53,54]
8.	Visualization and Interaction	[39,46,47,51–54]

The framework keeps SPC configurable by domain: heritage workloads skew toward micro-LoD, provenance, and uncertainty; city-scale planning favors coverage, rule checks, and scenario evaluation; AEC/FM emphasizes typed assets and scan-to-BIM consistency (Section 4.4). Performance/accuracy trade-offs and recommended levers are summarized in Section 4.6. The organizing principle is to treat the SPC data model as the backbone, into which geometric and learned signals are injected and upon which rules/ontologies operate—yielding pipelines that are auditable, updatable, and application-aware.

6. Remaining Challenges and Future Directions

Despite progress, the corpus highlights persistent obstacles that limit SPC adoption at scale. To answer RQ6, we first synthesize observed challenges in our corpus. We then outline mitigation strategies and forward directions. Cross-references point to the technical sections where relevant trade-offs and levers are detailed (Sections 4.3–4.6).

6.1. Key Challenges (Evidence from the Corpus)

Data Quality and Computational Demands—Noise [15], uneven density [51], mixed radiometry, and occlusions pose critical challenges in multi-source fusion, where heterogeneous datasets and registration residuals create conflicting feature vectors that confuse

standard classifiers. This degrades feature stability and segmentation/classification performance, raising false positives, causing failed merges, and yielding brittle descriptors [44,45] (Sections 4.3.3 and 4.3.4). At scale, memory and I/O become bottlenecks; voxelization/tiling improves access locality but shifts pressure to indexing and storage [43] (Section 4.5.3). Reported runtimes confirm sensitivity to scene complexity, data layout, and implementation choices (Section 4.6). The tension is clear: higher fidelity aids recognition but increases compute and storage costs.

Segmentation and Classification Limits—Unsupervised pipelines reliably recover dominant structural elements (e.g., planar components) but struggle with complex/non-planar geometry and heavy clutter [38,45]; supervised models are constrained by labeled-data scarcity in specialized domains (e.g., heritage) [39]. Several studies note mislabeling driven by repetitive patterns and density variation—over-segmentation in clutter and over-merging in sparse regions—and the difficulty of integrating semantics with geometry at decision time [44] (Sections 4.3.4, 4.3.5 and 4.6). Ontology rules improve consistency, but long-tail classes remain challenging without additional signals or priors.

Coverage of Application Settings—SPC validation was found to mostly concentrate on indoor/built contexts, with fewer studies evaluating environmental or city-scale regions [15]. These are likely under-represented because defining rigid semantic ontologies for unstructured vegetation/corridor scale is significantly more complex than for the regular, parametric geometries found in the built environment (AEC/heritage). However, this limits generalization claims and makes domain adaptation a priority (Section 4.4).

Temporal Evolution (toward 4D SPC)—While the SPC definition accommodates provenance/versioning (Section 4.3.6), few papers operationalize sustained temporal updates (e.g., stable identifications, deltas, temporal queries) [15]. Managing change (registration drift, re-segmentation, label stability) and storing/querying histories at scale remain open problems with clear payoffs for monitoring and asset management.

Bridging to Parametric 3D Models—Where downstream workflows require surfaces/solids (CAD/BIM/CityGML), robust, automatable "SPC → parametric" export is still limited. Primitive fitting can under-fit small deviations; implicit meshing may over-smooth detail (Section 4.3.6). Preserving SPC semantics/provenance through export—and enabling a round-trip path—is rarely addressed end-to-end.

Usability/Queries Accessibility—Current SPC access layers are largely SQL/SPARQL over relational/graph stores (Section 4.5.3). These are powerful but demand technical fluency; end-user-oriented query interfaces are uncommon. Lowering the barrier without sacrificing schema awareness and explainability is an adoption gap (Section 4.5.4).

Data Acquisition Evolution—Handheld MLS and other agile platforms appear but are not yet systematically benchmarked across SPC phases; solid-state LiDAR and tightly fused RGB-D/IMU stacks are emergent. Calibrating, geo-referencing, and fusing these sources into stable SPCs—without destabilizing density/attribute comparability across epochs—remain practical challenges [47]; (Sections 4.3.1 and 4.3.2).

6.2. Mitigation Strategies and Forward Directions

Below, proposals are grouped by SPC levers. Items marked (*beyond corpus*) extend beyond what our corpus explicitly demonstrated but reflect feasible, near-term directions.

6.2.1. Density/LoD and Pre-Processing at Scale

- LoD-aware Organization: Alignment of octree/tiling with SPC tiers (L0/L1/L2) so that heavy stages operate on bounded subsets and reprocessing is localized (Sections 4.3.6 and 4.5.3).

- Uniformization and Normalization: Using density-aware neighborhoods and controlled subsampling to stabilize descriptors across sensors (Sections 4.3.3 and 4.3.4).
- Computational Improvements (beyond corpus): Leveraging GPU acceleration, parallel frameworks, and compression to reduce wall time and I/O.
- Dynamic LoD (beyond corpus): Adapting point budgets to view distance/object importance to cap runtime during interaction.
- Learned Simplification (beyond corpus): Applying learning-based denoising (e.g., PointCleanNet [92]) and robust resampling (e.g., farthest-point sampling (FPS) [93]) to cut noise/redundancy while preserving morphology.

6.2.2. Segmentation and Classification Robustness

- Two-stage Pipelines: Over-segmenting with unsupervised methods, then refining with supervised and/or knowledge-based classification to reduce label noise (Sections 4.3.4 and 4.3.5).
- Rule-constrained Learning: Combining ML with ontology checks (e.g., reject "door with no adjacent wall") to stabilize rare classes without larger training sets (Sections 4.3.5 and 4.6).
- Context Encoders (beyond corpus): Exploring graph neural encoders over RAGs and self-supervised pretraining on unlabeled clouds to cut labeled-data needs [94].
- Synthetic/Active Data (beyond corpus): Generating targeted synthetic sets and using active learning to prioritize informative annotations [95].

6.2.3. Domain Transfer and Coverage

- Broader Domains: Encoding more domain-specific priors (vegetation strata, roadway topology, hydrologic rules) and exposing them as constraints in SPC rules.
- Geometric Reasoning (beyond corpus): Advances in AEC and environmental geometry-centered applications, such as optimization models for prefabricated construction and carbon efficiency [96–99] rely on precise geometric 3D ratios (e.g., area-to-perimeter, height-to-area). SPC frameworks can operationalize these for existing assets by automating the extraction of measurable morphological drivers from as-built data, directly feeding robotic 3D printing and renovation decision support systems.
- Domain Adaptation (beyond corpus): Transferring knowledge from labeled data of a domain to another with feature alignment or fine-tuning [100].
- Minimal Labeling (beyond corpus): Using few-shot/active strategies to bootstrap novel settings with limited ground truth [95].

6.2.4. Temporal SPC (4D)

- Operationalizing 4D Updates: To prioritize delta persistence over full rebuilds, 4D SPCs require object-level tracking via: (1) persistent IDs for stability across scans; (2) storing updates as timestamped relation triples in the knowledge graph (e.g., 'Wall_04 hasCondition at Time_T2') rather than duplicating geometry; and (3) employing temporal predicates (e.g., becameDamaged) to query state evolution.
- Spatio-temporal Indices (beyond corpus): Employing time-aware octrees/k-d trees for out-of-core 4D queries.
- Lightweight Temporal Models (beyond corpus): Applying sequence encoders (e.g., Long Short-Term Memory/recurrent neural networks) for change detection [101,102].

6.2.5. SPC → Geometry-Centric Exports

- Hybrid Surfacing: Fitting robust primitives for regular parts and triangulating residuals locally [103]; carrying SPC identifiers, roles, residuals, and uncertainties into CAD/BIM attributes to maintain provenance trail (Section 4.3.6).

- Procedural Hints (beyond corpus): Seeding shape grammars from SPC relations; exploring learned reconstruction where repeatable; preserving traceability [104].

6.2.6. Query UX and Explainability

- Schema-aware Presets: Shipping parameterized SQL/SPARQL snippets (i.e., "doors on walls," "low-confidence patches") to semantic styling in viewers (Section 4.5.4).
- Natural Language (NL) → Query (beyond corpus): Compiling NL intents into SPC-aware SQL/SPARQL via pre-trained models to broaden access [105].
- Visual Querying (beyond corpus): Integrating region select/filter with live previews; exposing "why this result" via rule hits and score overlays [106].

6.2.7. Orchestration and Agents

- Module Contracts: Encapsulating modules (pre-process/segment/classify/reason stages) with typed I/O and side-effect logging to enable restartable runs (Section 4.3).
- Artificial Intelligence (AI) Integration (beyond corpus): Introducing autonomous AI task agents to plan SPCs (e.g., optimizing LoD choices, scheduling consistency checks, and resolving contradictions).
- Planner Strategies (beyond corpus): Employing behavior trees for transparent control; exploring Reinforcement Learning (RL) where feedback signals are clear [107]; and supporting federated learning for privacy-sensitive deployments [108].

6.2.8. Sensors and Fusion

- Practical Calibration/Fusion: Investigating emerging sensors (handheld MLS, solid-state), prioritizing repeatable calibration, referencing control (GCPs/trajectories), and attribute normalization before fusion (Sections 4.3.1 and 4.3.2).
- SLAM and Completion (beyond corpus): Applying real-time SLAM for pose [68]; using DL completion models to mitigate occlusions/sparsity and recording provenance.

6.2.9. Minimum Reporting Checklist

Finally, to address the fragmentation and reproducibility challenges identified in this review, and to facilitate cross-study comparison, we propose a minimum reporting checklist for future SPC research. We recommend that authors explicitly document:

- **Definition Scope:** The specific SPC label used and the model's scope (Geometry, Semantics, Topology).
- **Dataset and Acquisition:** Sensor modality, point density, and scene characteristics.
- **Processing and Fusion:** Pre-processing/fusion strategy and software provenance.
- **Knowledge Integration:** Ontology/schema, relation types, and update strategy.
- **Performance Metrics:** Quantitative baselines including registration error, segmentation accuracy, query success rates/response time, and storage footprint.

7. Conclusions

The rapid advancements in remote sensing technologies have revolutionized data acquisition, delivering highly detailed point clouds essential for digital spatial applications. Processing this data has been a focal area of recent research, aiming to optimize and automate stages of the point cloud processing workflow. Synthesizing 58 publications, this Systematic Literature Review examined the evolving "Smart Point Cloud" (SPC) framework, we clarify usage and terminology, adopt an operative definition, and consolidate methods and tools across domains. We define an SPC as a queryable, multi-scale, ontology-linked, provenance-aware point cloud data model that augments geometry with

semantics and explicit relations/topology, and supports rule/ML-driven automation and incremental updates.

Evidence from cultural heritage, urban planning, and AEC/FM shows tangible benefits: semantic queries, rule checks, and auditable updates—shifting deliverables from geometry-only products to decision-ready assets. Methodologically, the field is algorithm-agnostic yet patterned. We distill a modular eight-module framework—acquisition and pre-processing; registration and fusion; hierarchical segmentation; knowledge-based classification (including ontology reasoning); SPC data modeling; storage and indexing; access and reasoning; and visualization—so future implementations can tune pipelines to scale, accuracy targets, and compute budgets.

Cross-study trends offer actionable guidance: (i) higher and more uniform density stabilizes features but increases storage/I/O; (ii) neuro-symbolic flows (ML/DL + ontology constraints) improve consistency on long-tail classes; and (iii) LoD indexing and out-of-core access, coupled with compiled/GPU kernels, keep runtimes tractable. Persistent challenges include data quality at scale, domain transfer beyond indoor/built scenes, temporal (4D) SPCs with versioned tiers and incremental updates, semantics-preserving SPC→parametric exports, usable query user experience (UX) (NL-to-query, visual querying), and robust calibration/fusion for emerging sensors.

Limitations of this review include the heterogeneity in datasets, metrics, and hardware across the corpus, which may preclude quantitative meta-analysis. Additionally, the restriction to English-language publications and specific academic databases may have excluded relevant regionally published works. Finally, while our keyword strategy prioritized 'Smart/Rich/Intelligent' terminology to trace the field's evolution, we acknowledge that relevant work may exist under adjacent labels (e.g., ontology, semantics), as evidenced in our "*Indirect*" corpus and co-occurring terms. Future reviews could expand searches to these functional synonyms for techniques outside the explicit SPC nomenclature. Furthermore, while bolstered by co-author verification on a subset, reliance on a single primary reviewer remains a limitation.

To improve future discoverability and comparability, we recommend lightweight reporting standards for SPC research, covering explicit labeling, definition, and keywords aligned with our findings, and clear acquisition parameters, semantic scope, and performance metrics. With these practices and the proposed modular framework, Smart Point Clouds can evolve into a durable backbone for analytics, monitoring, and lifecycle decision support across domains.

Author Contributions: Conceptualization, M.H.S., A.S. and S.L.; methodology, software, validation, formal analysis, and investigation, M.H.S.; resources, A.S. and S.L.; data curation and writing—original draft preparation, M.H.S.; writing—review and editing, A.S. and S.L.; visualization, M.H.S.; supervision, project administration, and funding acquisition, A.S. and S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC), grant number RGPIN-2020-05857, and the Smart Campus Integrated Platform Development Alliance project with FuseForward. Funders had no role in study design; collection, analyses, or interpretation of data; writing of the manuscript; or the decision to publish the results.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: The authors acknowledge the support of NSERC and the Smart Campus Integrated Platform Development Alliance project with FuseForward.

Conflicts of Interest: The authors declare no conflicts of interest. Funders had no role in the study design; collection, analyses, or interpretation of data; manuscript writing; or the decision to publish.

Appendix A

Table A1. Initial list of 20 candidate keywords considered during scoping.

Query Terms (Keywords)		
Smart Point Cloud(s)	Three-dimensional Intelligent Point Data	Smart Cloud-of-Points
Point Cloud Smart Mapping	Intelligent Point Cloud(s)	Smart Point Clouds
Smart 3D Laser Modeling	Hexagon Smart Point Cloud	Smart 3D Point Cloud
Smart Point Cloud Workflow	Point Cloud Processing	Smart Point Cloud
Scan-to-BIM	Object Labeling	Smart Urban Reconstruction
Rich Point Cloud(s)	Automatic Extraction	Semantic Enrichment
Semantic Segmentation	Smart 3D Point Cloud Classification	

Table A2. Boolean search query strings constructed and applied across selected databases using the three main keyword themes.

Search Query Strings
(“SMART” AND “POINT” AND “CLOUD”) OR (“SMART” AND “POINT” AND “CLOUDS”) OR (“SMART” AND “POINTS” AND “CLOUD”) OR (“SMART” AND “POINTCLOUD”)
(“INTELLIGENT” AND “POINT” AND “CLOUD”) OR (“INTELLIGENT” AND “POINT” AND “CLOUDS”) OR (“INTELLIGENT” AND “POINTS” AND “CLOUD”)
(“RICH” AND “POINT” AND “CLOUD”) OR (“RICH” AND “POINT” AND “CLOUDS”) OR (“RICH” AND “POINTS” AND “CLOUD”)

Table A3. Summary of search results by database and search phase (pre-deduplication).

Source	URL	Step 1 Title Only	Step 2 Full Text [3]	Total
Scopus [1]	https://www.scopus.com/ [29]	141	n/a	141
Web of Science [1]	https://apps.webofknowledge.com/ [30]	98	n/a	98
Google Scholar [1,2]	https://scholar.google.com/ [31]	215	154	369
Total		454	154	608

[1] Unbounded Retrieval: Every result returned by the engine was manually verified against eligibility criteria, rather than relying on relevance ranking or “Top-N” sampling. [2] Privacy Settings: Searches were executed in private browsing sessions (from Toronto, Canada) to minimize personalization bias. [3] Search Settings: Full-text search was executed only in Google Scholar via Advanced Search, using the “with the exact phrase” function for Smart Point Cloud and the “where my words occur” selection set to “anywhere in the article”.

Table A4. Final list of studies included in the analysis (n = 58), with category labels.

Category	Count	Studies
Direct	22	[13–15,36–54]
Indirect	22	[82,91,109–128]
New Use	14	[3,33–35,55–64]
Total	58	

References

1. Yang, F.; Li, X.; Shen, J. Nested Architecture Search for Point Cloud Semantic Segmentation. *IEEE Trans. Image Process.* **2023**, *32*, 2889–2900. [CrossRef]
2. Gharineiat, Z.; Tarsha Kurdi, F.; Campbell, G. Review of Automatic Processing of Topography and Surface Feature Identification LiDAR Data Using Machine Learning Techniques. *Remote Sens.* **2022**, *14*, 4685. [CrossRef]

3. El-Din Fawzy, H. 3D Laser Scanning and Close-Range Photogrammetry for Buildings Documentation: A Hybrid Technique towards a Better Accuracy. *Alex. Eng. J.* **2019**, *58*, 1191–1204. [CrossRef]
4. Moyano, J.; Nieto-Julián, J.E.; Fernández-Alconchel, M.; Oreni, D.; Estévez-Pardal, R. Analysis and Precision of Light Detection and Ranging Sensors Integrated in Mobile Phones as a Framework for Registration of Ground Control Points for Unmanned Aerial Vehicles in the Scanning Technique for Building Information Modelling in Archaeological Sites. *Drones* **2023**, *7*, 477. [CrossRef]
5. Di Stefano, F.; Chiappini, S.; Gorreja, A.; Balestra, M.; Pierdicca, R. Mobile 3D Scan LiDAR: A Literature Review. *Geomat. Nat. Hazards Risk* **2021**, *12*, 2387–2429. [CrossRef]
6. Hyyppä, E.; Hyyppä, J.; Hakala, T.; Kukko, A.; Wulder, M.A.; White, J.C.; Pyörälä, J.; Yu, X.; Wang, Y.; Virtanen, J.P.; et al. Under-Canopy UAV Laser Scanning for Accurate Forest Field Measurements. *ISPRS J. Photogramm. Remote Sens.* **2020**, *164*, 41–60. [CrossRef]
7. Lopac, N.; Jurdana, I.; Brnelić, A.; Krljan, T. Application of Laser Systems for Detection and Ranging in the Modern Road Transportation and Maritime Sector. *Sensors* **2022**, *22*, 5946. [CrossRef]
8. Ulvi, A. Documentation, Three-Dimensional (3D) Modelling and Visualization of Cultural Heritage by Using Unmanned Aerial Vehicle (UAV) Photogrammetry and Terrestrial Laser Scanners. *Int. J. Remote Sens.* **2021**, *42*, 1994–2021. [CrossRef]
9. Wang, Q.; Kim, M.K. Applications of 3D Point Cloud Data in the Construction Industry: A Fifteen-Year Review from 2004 to 2018. *Adv. Eng. Inform.* **2019**, *39*, 306–319. [CrossRef]
10. Altuntas, C. Review of Scanning and Pixel Array-Based LiDAR Point-Cloud Measurement Techniques to Capture 3D Shape or Motion. *Appl. Sci.* **2023**, *13*, 6488. [CrossRef]
11. Xia, S.; Chen, D.; Wang, R.; Li, J.; Zhang, X. Geometric Primitives in LiDAR Point Clouds: A Review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 685–707. [CrossRef]
12. Chitta, A. *Replacing Objects in Point Cloud Stream with Real-Time Meshes Using Semantic Segmentation*; Blekinge Institute of Technology: Karlskrona, Sweden, 2024.
13. Yang, S.; Hou, M.; Shaker, A.; Li, S. Modeling and Processing of Smart Point Clouds of Cultural Relics with Complex Geometries. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 617. [CrossRef]
14. Poux, F.; Neuville, R.; Billen, R. Point cloud classification of tesserae from terrestrial laser data combined with dense image matching for archaeological information extraction. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *IV-2/W2*, 203–211. [CrossRef]
15. Poux, F.; Billen, R. A Smart Point Cloud Infrastructure for Intelligent Environments. In *Laser Scanning*; CRC Press: Boca Raton, FL, USA, 2019; pp. 127–149. ISBN 9781351018869.
16. Yang, T.; Zou, Y.; Yang, X.; del Rey Castillo, E. Domain Knowledge-Enhanced Region Growing Framework for Semantic Segmentation of Bridge Point Clouds. *Autom. Constr.* **2024**, *165*, 105572. [CrossRef]
17. Sohail, S.S.; Himeur, Y.; Kheddar, H.; Amira, A.; Fadli, F.; Atalla, S.; Copiaco, A.; Mansoor, W. Advancing 3D Point Cloud Understanding through Deep Transfer Learning: A Comprehensive Survey. *Inf. Fusion* **2025**, *113*, 102601. [CrossRef]
18. Lokugam Hewage, C.N.; Laefer, D.F.; Vo, A.-V.; Le-Khac, N.-A.; Bertolotto, M. Scalability and Performance of LiDAR Point Cloud Data Management Systems: A State-of-the-Art Review. *Remote Sens.* **2022**, *14*, 5277. [CrossRef]
19. Ding, Z.; Sun, Y.; Xu, S.; Pan, Y.; Peng, Y.; Mao, Z. Recent Advances and Perspectives in Deep Learning Techniques for 3D Point Cloud Data Processing. *Robotics* **2023**, *12*, 100. [CrossRef]
20. Jancewicz, K.; Porębna, W. Point Cloud Does Matter. Selected Issues of Using Airborne LiDAR Elevation Data in Geomorphometric Studies of Rugged Sandstone Terrain under Forest—Case Study from Central Europe. *Geomorphology* **2022**, *412*, 108316. [CrossRef]
21. Xu, Y.; Stilla, U. Toward Building and Civil Infrastructure Reconstruction from Point Clouds: A Review on Data and Key Techniques. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2857–2885. [CrossRef]
22. Che, E.; Jung, J.; Olsen, M.J. Object Recognition, Segmentation, and Classification of Mobile Laser Scanning Point Clouds: A State of the Art Review. *Sensors* **2019**, *19*, 810. [CrossRef]
23. Abreu, N.; Pinto, A.; Matos, A.; Pires, M. Procedural Point Cloud Modelling in Scan-to-BIM and Scan-vs-BIM Applications: A Review. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 260. [CrossRef]
24. Shaheen, N.; Shaheen, A.; Ramadan, A.; Hefnawy, M.T.; Ramadan, A.; Ibrahim, I.A.; Hassanein, M.E.; Ashour, M.E.; Flouty, O. Appraising Systematic Reviews: A Comprehensive Guide to Ensuring Validity and Reliability. *Front. Res. Metr. Anal.* **2023**, *8*, 1268045. [CrossRef]
25. Xiao, Y.; Watson, M. Guidance on Conducting a Systematic Literature Review. *J. Plan. Educ. Res.* **2019**, *39*, 93–112. [CrossRef]
26. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews. *BMJ* **2021**, *372*, 71. [CrossRef]
27. Google. Available online: https://www.google.com/ (accessed on 12 October 2024).

28. DePoy, E. *Introduction to Research-e-Book: Understanding and Applying Multiple Strategies*, 7th ed.; Elsevier Health Sciences: Amsterdam, The Netherlands, 2024; ISBN 9780443109416.
29. Scopus. Available online: https://www.scopus.com/ (accessed on 18 June 2025).
30. Web of Science | Clarivate. Available online: https://apps.webofknowledge.com/ (accessed on 18 June 2025).
31. Google Scholar. Available online: https://scholar.google.com/ (accessed on 18 June 2025).
32. Chigbu, U.E.; Atiku, S.O.; Du Plessis, C.C. The Science of Literature Reviews: Searching, Identifying, Selecting, and Synthesising. *Publications* **2023**, *11*, 2. [CrossRef]
33. Nir, E.; Capeluto, G. Smart Cloud-of-Points Model: Point-Based Digital Media and Tools for Architectural Design. In Proceedings of the eCAADe 2005, Lisbon, Portugal, 21–24 September 2005; pp. 687–694.
34. Nir, E.; Capeluto, G. On Design Continuity with Smart Cloud of Points. In Proceedings of the eCAADe 2006, Volos, Greece, 6–9 September 2006; pp. 794–797.
35. Nebiker, S.; Bleisch, S.; Christen, M. Rich Point Clouds in Virtual Globes—A New Paradigm in City Modeling? *Comput. Environ. Urban Syst.* **2010**, *34*, 508–517. [CrossRef]
36. Poux, F.; Billen, R. Smart Point Cloud: Toward an Intelligent Documentation of Our World. In Proceedings of the Professionnal Conference (PCON) 2015, Liège, Belgium, 17 December 2015; pp. 1–11.
37. Poux, F.; Hallot, P.; Neuville, R.; Billen, R. Smart point cloud: Definition and remaining challenges. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *IV-2/W1*, 119–127. [CrossRef]
38. Poux, F. Point Clouds, Segments, Semantics and Automation. In Proceedings of the Notre-Dame Scientific Project (CNRS)—State of Play and Prospects, Paris, France, 20 October 2020; Notre-Dame Scientific Project (CNRS): Paris, France, 2020.
39. Poux, F. *The Smart Point Cloud: Structuring 3D Intelligent Point Data, ULiège*; Université de Liège: Liège, Belgium, 2019.
40. Poux, F.; Neuville, R.; Hallot, P.; Billen, R. Model for semantically rich point cloud data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *IV-4/W5*, 107–115. [CrossRef]
41. Billen, R.; Poux, F.; Van Ruymbeke, M. Smart Point Clouds for Information Modelling: Application in Cultural Heritage. In Proceedings of the Annual Netherlands Centre for Geodesy and Geo-Informatics (NCG) Symposium, Delft, The Netherlands, 2 November 2017; Netherlands Centre for Geodesy and Geo-Informatics (NCG): Delft, The Netherlands, 2017.
42. Poux, F.; Neuville, R.; Hallot, P.; Billen, R. Point Clouds as an Efficient Multiscale Layered Spatial Representation. In Proceedings of the Eurographics Workshop on Urban Data Modelling and Visualisation (UDMV), Liège, Belgium, 8 December 2016, Biljecki, F., Tourre, V., Eds.; Eurographics: Eindhoven, The Netherlands, 2016; pp. 31–36.
43. Balado, J.; van Oosterom, P.; Díaz-Vilariño, L.; Meijers, M. Mathematical Morphology Directly Applied to Point Cloud Data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *168*, 208–220. [CrossRef]
44. Poux, F.; Ponciano, J.J. Self-learning ontology for instance segmentation of 3D indoor point cloud. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B2-2020*, 309–316. [CrossRef]
45. Poux, F.; Billen, R. Voxel-Based 3D Point Cloud Semantic Segmentation: Unsupervised Geometric and Relationship Featuring vs. Deep Learning Methods. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 213. [CrossRef]
46. Zheng, M. *Classification of Point Clouds of Urban Scenes Exploiting a Hierarchy of Point Densities*; Delft University of Technology: Delft, The Netherlands, 2019.
47. van Bochove, D.P. *Combining MLS & ALS Point Cloud Data*; Utrecht University: Utrecht, The Netherlands, 2019.
48. Joosten, F. Map Supported Point Cloud Registration-a Method for Creation of a Smart Point Cloud. Master's Thesis, Delft University of Technology, Delft, The Netherlands, Utrecht University, Delft, The Netherlands, Wageningen University, Delft, The Netherlands, University of Twente, Delft, The Netherlands, 2018.
49. Kotoula, L. *The Smart Point Cloud Framework to Detect Pipelines Using Raw Point Cloud Generated from Panoramic Images*; Delft University of Technology: Delft, The Netherlands, 2018.
50. Zheng, M.; Lemmens, M.; Van Oosterom, P. Classification of mobile laser scanning point clouds from height features. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *XLII-2/W7*, 321–325. [CrossRef]
51. Poux, F.; Neuville, R.; Nys, G.A.; Billen, R. 3D Point Cloud Semantic Modelling: Integrated Framework for Indoor Spaces and Furniture. *Remote Sens.* **2018**, *10*, 1412. [CrossRef]
52. Kharroubi, A.; Hajji, R.; Billen, R.; Poux, F. Classification and integration of massive 3D points clouds in a virtual reality (VR) environment. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-2-W17*, 165–171. [CrossRef]
53. Poux, F.; Neuville, R.; Van Wersch, L.; Nys, G.A.; Billen, R. 3D Point Clouds in Archaeology: Advances in Acquisition, Processing and Knowledge Integration Applied to Quasi-Planar Objects. *Geosciences* **2017**, *7*, 96. [CrossRef]
54. Poux, F.; Neuville, R.; Hallot, P.; Van Wersch, L.; Jancsó, A.L.; Billen, R. Digital investigations of an archaeological smart point cloud: A real time web-based platform to manage the visualisation of semantical queries. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *XLII-5-W1*, 581–588. [CrossRef]
55. Koshan, Y.; Manish, R.; Joseph, M.; Habib, A. Alternative lidar technologies for stockpile monitoring and reporting. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *XLVIII-1/W2-2023*, 649–656. [CrossRef]

56. Gandikota, D.M.; Gladkova, T.; Tran, K.A.; Bapat, S.; Richkus, J.; Arnold, D.J. AI Augmentation to Remote Sensing Imagery in Forestry Conservation and Restoration for Increased Responsive Capabilities. In Proceedings of the 2022 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Washington, DC, USA, 11–13 October 2022; Institute of Electrical and Electronics Engineers Inc. (IEEE): Washington, DC, USA, 2022; pp. 1–16.
57. Aviv, D.; Guo, H.; Middel, A.; Meggers, F. Evaluating Radiant Heat in an Outdoor Urban Environment: Resolving Spatial and Temporal Variations with Two Sensing Platforms and Data-Driven Simulation. *Urban Clim.* **2021**, *35*, 100745. [CrossRef]
58. Aviv, D. *Design for Heat Transfer: Formal and Material Strategies to Leverage Thermodynamics in the Built Environment*; Princeton University: Princeton, BC, Canada, 2020.
59. Piniotis, G.; Soile, S.; Bourexis, F.; Tsakiri, M.; Ioannidis, C. Experimental assessment of 3D narrow space mapping technologies. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B2-2020*, 149–156. [CrossRef]
60. El-Din Fawzy, H.; Basha, A.M.; Botross, M.N. Estimating a Mathematical Formula of Soil Erosion under the Effect of Rainfall Simulation by Digital Close Range Photogrammetry Technique. *Alex. Eng. J.* **2020**, *59*, 5079–5097. [CrossRef]
61. Fawzy, H.E.-D.; Din, E.S. El Monitoring and Assessment of Road Cracks and Potholes Using Digital Close Range Photogrammetry 3D Reconstruction Technique. *Int. J. Civ. Eng. Technol.* **2020**, *11*, 119–128. [CrossRef]
62. Fawzy, H.E.-D. Study the Accuracy of Digital Close Range Photogrammetry Technique Software as a Measuring Tool. *Alex. Eng. J.* **2019**, *58*, 171–179. [CrossRef]
63. Konstantakis, M.; Trichopoulos, G.; Aliprantis, J.; Gavogiannis, N.; Karagianni, A.; Parthenios, P.; Serraos, K.; Caridakis, G. An Improved Approach for Generating Digital Twins of Cultural Spaces through the Integration of Photogrammetry and Laser Scanning Technologies. *Digital* **2024**, *4*, 215–231. [CrossRef]
64. Wang, D. *Quantification of Single-Tree Structure in Mountain Forests Using Terrestrial Laser Scanning*; Technische Universität Wien: Vienna, Austria, 2018.
65. Gomez, C. *Point Cloud Technologies for Geomorphologists*; Springer Textbooks in Earth Sciences, Geography and Environment; Springer International Publishing: Cham, Switzerland, 2022; ISBN 978-3-031-10974-4.
66. Jasim, A.N.; Fourati, L.C.; Albahri, O.S. Evaluation of Unmanned Aerial Vehicles for Precision Agriculture Based on Integrated Fuzzy Decision-Making Approach. *IEEE Access* **2023**, *11*, 75037–75062. [CrossRef]
67. Yan, G.; Luo, Z.; Liu, Z.; Li, Y. SensorX2car: Sensors-to-Car Calibration for Autonomous Driving in Road Scenarios. *arXiv* **2023**, arXiv:2301.07279.
68. Cattaneo, D.; Vaghi, M.; Valada, A. LCDNet: Deep Loop Closure Detection and Point Cloud Registration for LiDAR SLAM. *IEEE Trans. Robot.* **2022**, *38*, 2074–2093. [CrossRef]
69. Jiang, B.; Shen, S. Contour Context: Abstract Structural Distribution for 3D LiDAR Loop Detection and Metric Pose Estimation. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; Institute of Electrical and Electronics Engineers Inc. (IEEE): London, UK, 2023; pp. 8386–8392.
70. Gan, V.J.L.; Hu, D.; Wang, Y.; Zhai, R. Automated Indoor 3D Scene Reconstruction with Decoupled Mapping Using Quadruped Robot and LiDAR Sensor. *Comput. Civ. Infrastruct. Eng.* **2025**, *40*, 2209–2226. [CrossRef]
71. Bai, X.; Luo, Z.; Zhou, L.; Chen, H.; Li, L.; Hu, Z.; Fu, H.; Tai, C.-L. PointDSC: Robust Point Cloud Registration Using Deep Spatial Consistency. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; Institute of Electrical and Electronics Engineers Inc. (IEEE): Nashville, TN, USA, 2021; pp. 15854–15864.
72. Lu, W.; Wan, G.; Zhou, Y.; Fu, X.; Yuan, P.; Song, S. DeepVCP: An End-to-End Deep Neural Network for Point Cloud Registration. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; Institute of Electrical and Electronics Engineers Inc. (IEEE): Seoul, Republic of Korea, 2019; pp. 12–21.
73. Ao, S.; Hu, Q.; Yang, B.; Markham, A.; Guo, Y. SpinNet: Learning a General Surface Descriptor for 3D Point Cloud Registration. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; Institute of Electrical and Electronics Engineers Inc. (IEEE): Nashville, TN, USA, 2021; pp. 11753–11762.
74. Li, P.; Wang, R.; Wang, Y.; Tao, W. Evaluation of the ICP Algorithm in 3D Point Cloud Registration. *IEEE Access* **2020**, *8*, 68030–68048. [CrossRef]
75. Zhao, W.; Zhang, D.; Li, D.; Zhang, Y.; Ling, Q. Optimized GICP Registration Algorithm Based on Principal Component Analysis for Point Cloud Edge Extraction. *Meas. Control* **2024**, *57*, 77–89. [CrossRef]
76. Shen, Y.; Zhang, B.; Wang, J.; Zhang, Y.; Wu, Y.; Chen, Y.; Chen, D. MI-NDT: Multiscale Iterative Normal Distribution Transform for Registering Large-Scale Outdoor Scans. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5705513. [CrossRef]
77. Cheng, L.; Chen, S.; Liu, X.; Xu, H.; Wu, Y.; Li, M.; Chen, Y. Registration of Laser Scanning Point Clouds: A Review. *Sensors* **2018**, *18*, 1641. [CrossRef]
78. Ismail, M.H.; Shaker, A.; Li, S. Developing complete urban digital twins in busy environments: A framework for facilitating 3D model generation from multi-source point cloud data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *XLVIII-1-W2-2023*, 7–14. [CrossRef]

79. Shao, Y.; Tong, G.; Peng, H. Mining Local Geometric Structure for Large-Scale 3D Point Clouds Semantic Segmentation. *Neurocomputing* **2022**, *500*, 191–202. [CrossRef]
80. Alshawabkeh, Y. Linear Feature Extraction from Point Cloud Using Color Information. *Herit. Sci.* **2020**, *8*, 1–13. [CrossRef]
81. Liu, H.; Guo, Y.; Ma, Y.; Lei, Y.; Wen, G. Semantic Context Encoding for Accurate 3D Point Cloud Segmentation. *IEEE Trans. Multimed.* **2021**, *23*, 2045–2055. [CrossRef]
82. Yang, S.; Hou, M.; Li, S. Three-Dimensional Point Cloud Semantic Segmentation for Cultural Heritage: A Comprehensive Review. *Remote Sens.* **2023**, *15*, 548. [CrossRef]
83. Quan, S.; Yang, J. Compatibility-Guided Sampling Consensus for 3-D Point Cloud Registration. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 7380–7392. [CrossRef]
84. Zou, Y.; Weinacker, H.; Koch, B. Towards Urban Scene Semantic Segmentation with Deep Learning from LiDAR Point Clouds: A Case Study in Baden-Württemberg, Germany. *Remote Sens.* **2021**, *13*, 3220. [CrossRef]
85. Liao, L.; Tang, S.; Liao, J.; Li, X.; Wang, W.; Li, Y.; Guo, R. A Supervoxel-Based Random Forest Method for Robust and Effective Airborne LiDAR Point Cloud Classification. *Remote Sens.* **2022**, *14*, 1516. [CrossRef]
86. Zhang, J.; Zhao, X.; Chen, Z.; Lu, Z. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. *IEEE Access* **2019**, *7*, 179118–179133. [CrossRef]
87. Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M. Deep Learning for 3D Point Clouds: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4338–4364. [CrossRef]
88. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic Graph CNN for Learning on Point Clouds. *ACM Trans. Graph.* **2019**, *38*, 1. [CrossRef]
89. Vinodkumar, P.K.; Karabulut, D.; Avots, E.; Ozcinar, C.; Anbarjafari, G. A Survey on Deep Learning Based Segmentation, Detection and Classification for 3D Point Clouds. *Entropy* **2023**, *25*, 635. [CrossRef]
90. Mehranfar, M.; Vega-Torres, M.A.; Braun, A.; Borrmann, A. Automated Data-Driven Method for Creating Digital Building Models from Dense Point Clouds and Images through Semantic Segmentation and Parametric Model Fitting. *Adv. Eng. Inform.* **2024**, *62*, 102643. [CrossRef]
91. Kharroubi, A.; Van Wersch, L.; Billen, R.; Poux, F. Tesserae3D: A benchmark for tesserae semantic segmentation in 3D point clouds. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *V-2-2021*, 121–128. [CrossRef]
92. Rakotosaona, M.J.; La Barbera, V.; Guerrero, P.; Mitra, N.J.; Ovsjanikov, M. PointCleanNet: Learning to Denoise and Remove Outliers from Dense Point Clouds. *Comput. Graph. Forum* **2019**, *39*, 185–203. [CrossRef]
93. Zhao, T.; Li, L.; Tian, T.; Ma, J.; Tian, J. APUNet: Attention-Guided Upsampling Network for Sparse and Non-Uniform Point Cloud. *Pattern Recognit.* **2023**, *143*, 109796. [CrossRef]
94. Qi, C.R. Deep Learning on 3D Data. In *3D Imaging, Analysis and Applications*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 513–566, ISBN 9783030440701.
95. Shao, F.; Luo, Y.; Liu, P.; Chen, J.; Yang, Y.; Lu, Y.; Xiao, J. Active Learning for Point Cloud Semantic Segmentation via Spatial-Structural Diversity Reasoning. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; Association for Computing Machinery, Inc.: Lisboa, Portugal, 2022; pp. 2575–2585.
96. Cai, G.; Liu, D.; Wu, Z. Multidimensional Algorithms-Based Carbon Efficiency Model of Building Geometric 3D Ratios for Prefabricated 3D Printing Design and Construction. *npj Clean Energy* **2025**, *1*, 5. [CrossRef]
97. Cai, G.; Sun, L.; Liu, D.; Xu, B.; Mo, Z. Potential of Indoor Room 3D Ratio in Reducing Carbon Emissions by Prefabricated Decoration in Chain Hotel Buildings via Multidimensional Algorithm Models for Robot In-Situ 3D Printing. *J. Build. Eng.* **2025**, *101*, 111757. [CrossRef]
98. Cai, G.; Lou, Y.; Lu, F. AI Enhancing Prefabricated Aesthetics and Low Carbon Coupled with 3D Printing in Chain Hotel Buildings from Multidimensional Neural Networks. *Sci. Rep.* **2025**, *15*, 13229. [CrossRef]
99. Cai, G.; Guo, X.; Su, Y. Prefabricated Building Construction in Materialization Phase as Catalysts for Hotel Low-Carbon Transitions via Hybrid Computational Visualization Algorithms. *Sci. Rep.* **2025**, *15*, 7734. [CrossRef] [PubMed]
100. Xing, B.; Ying, X.; Wang, R.; Yang, J.; Chen, T. Cross-Modal Contrastive Learning for Domain Adaptation in 3D Semantic Segmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Association for the Advancement of Artificial Intelligence Press: Washington, DC, USA, 2023; Volume 37, pp. 2974–2982.
101. Sun, X.; Wang, S.; Liu, M. A Novel Coding Architecture for Multi-Line LiDAR Point Clouds Based on Clustering and Convolutional LSTM Network. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 2190–2201. [CrossRef]
102. Knights, J.; Moghadam, P.; Ramezani, M.; Sridharan, S.; Fookes, C. InCloud: Incremental Learning for Point Cloud Place Recognition. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; Institute of Electrical and Electronics Engineers Inc.: Kyoto, Japan, 2022; pp. 8559–8566.
103. Wu, Q.; Liu, J.; Gao, C.; Wang, B.; Shen, G.; Li, Z. Improved RANSAC Point Cloud Spherical Target Detection and Parameter Estimation Method Based on Principal Curvature Constraint. *Sensors* **2022**, *22*, 5850. [CrossRef]

104. Chen, X.; Ravikumar, N.; Xia, Y.; Attar, R.; Diaz-Pinto, A.; Piechnik, S.K.; Neubauer, S.; Petersen, S.E.; Frangi, A.F. Shape Registration with Learned Deformations for 3D Shape Reconstruction from Sparse and Incomplete Point Clouds. *Med. Image Anal.* **2021**, *74*, 102228. [CrossRef]
105. Shen, L.; Shen, E.; Luo, Y.; Yang, X.; Hu, X.; Zhang, X.; Tai, Z.; Wang, J. Towards Natural Language Interfaces for Data Visualization: A Survey. *IEEE Trans. Vis. Comput. Graph.* **2023**, *29*, 3121–3144. [CrossRef]
106. Qi, J.; Lin, E.S.; Yok Tan, P.; Chun Man Ho, R.; Sia, A.; Olszewska-Guizzo, A.; Zhang, X.; Waykool, R. Development and Application of 3D Spatial Metrics Using Point Clouds for Landscape Visual Quality Assessment. *Landsc. Urban Plan.* **2022**, *228*, 104585. [CrossRef]
107. Rosas, F.G.; Hoeller, F.; Schneider, F.E. Autonomous Robot Exploration and Measuring in Disaster Scenarios Using Behavior Trees. In Proceedings of the 2020 IEEE 10th International Conference on Intelligent Systems (IS), Varna, Bulgaria, 28–30 August 2020; Institute of Electrical and Electronics Engineers Inc.: Varna, Bulgaria, 2020; pp. 469–474.
108. Zhou, Y.; Li, B.; Huang, K. An Object Detection Method Based on Heterogeneous Lidar Point Clouds Using Federated Learning. In Proceedings of the 3rd 2023 International Conference on Autonomous Unmanned Systems (3rd ICAUS 2023), Nanjing, China, 8–11 September 2023; Springer: Singapore, 2024; Volume 1176, pp. 248–259.
109. Zhang, J.; Wang, Z.; Wang, W. Application of Smart Modelling Framework for Traditional Wooden Architecture. *Buildings* **2024**, *14*, 2130. [CrossRef]
110. Yang, S.; Hou, M.; Fan, H. CityGML Grotto ADE for Modelling Niches in 3D with Semantic Information. *Herit. Sci.* **2024**, *12*, 135. [CrossRef]
111. Jeddoub, I.; Ballouch, Z.; Hajji, R.; Billen, R. Enriched Semantic 3D Point Clouds: An Alternative to 3D City Models for Digital Twin for Cities? In *Recent Advances in 3D Geoinformation Science: Proceedings of the 18th International 3D GeoInfo Conference, Munich, Germany, 12–14 September 2023*; Kolbe, T.H., Donaubauer, A., Beil, C., Eds.; Springer Nature: Berlin/Heidelberg, Germany, 2024; pp. 407–423. ISBN 978-3-031-43699-4.
112. Galanakis, D.; Maravelakis, E.; Pocobelli, D.P.; Vidakis, N.; Petousis, M.; Konstantaras, A.; Tsakoumaki, M. SVD-Based Point Cloud 3D Stone by Stone Segmentation for Cultural Heritage Structural Analysis—The Case of the Apollo Temple at Delphi. *J. Cult. Herit.* **2023**, *61*, 177–187. [CrossRef]
113. Zhao, J.; Hua, X.; Yang, J.; Yin, L.; Liu, Z.; Wang, X. A review of point cloud segmentation of architectural cultural heritage. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *X-1/W1-2023*, 247–254. [CrossRef]
114. Vijaywargiya, J.; Ramiya, A.M. Information Extraction System for Urban Planning and Governance Using LiDAR Based 3D Repository. *J. Spat. Sci.* **2024**, *69*, 203–223. [CrossRef]
115. Franzini, M.; Casella, V.M.; Monti, B. Assessment of Leica CityMapper-2 LiDAR Data within Milan's Digital Twin Project. *Remote Sens.* **2023**, *15*, 5263. [CrossRef]
116. Ogayar-Anguita, C.J.; López-Ruiz, A.; Rueda-Ruiz, A.J.; Segura-Sánchez, R.J. Nested Spatial Data Structures for Optimal Indexing of LiDAR Data. *ISPRS J. Photogramm. Remote Sens.* **2023**, *195*, 287–297. [CrossRef]
117. Yang, S.; Xu, S.; Huang, W. 3D Point Cloud for Cultural Heritage: A Scientometric Survey. *Remote Sens.* **2022**, *14*, 5542. [CrossRef]
118. Zhao, X.; Li, Q.; Zhang, M.; Wang, Q.; He, Z.; Lv, Y. Closely Adjacent Object Segmentation Based on Periphery-Restrictive Region Growing. *Meas. Sci. Technol.* **2022**, *33*, 125901. [CrossRef]
119. Murtiyoso, A.; Matrone, F.; Martini, M.; Lingua, A.; Grussenmeyer, P.; Pierdicca, R. Automatic training data generation in deep learning-aided semantic segmentation of heritage buildings. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2022**, *V-2–2022*, 317–324. [CrossRef]
120. Beil, C.; Kutzner, T.; Schwab, B.; Willenborg, B.; Gawronski, A.; Kolbe, T.H. Integration of 3D point clouds with semantic 3D city models—Providing semantic information beyond classification. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *VIII-4/W2-2021*, 105–112. [CrossRef]
121. He, Z.; Wang, H.; Zhao, X.; Zhang, S.; Tan, Z. Inward-Region-Growing-Based Accurate Partitioning of Closely Stacked Objects for Bin-Picking. *Meas. Sci. Technol.* **2020**, *31*, 125901. [CrossRef]
122. Poux, F.; Billen, R.; Kasprzyk, J.P.; Lefebvre, P.H.; Hallot, P. A Built Heritage Information System Based on Point Cloud Data: HIS-PC. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 588. [CrossRef]
123. Poux, F.; Mattes, C.; Kobbelt, L. Unsupervised segmentation of indoor 3D point cloud: Application to object-based classification. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIV-4/W1-2020*, 111–118. [CrossRef]
124. El-Mahgary, S.; Virtanen, J.P.; Hyyppä, H. A Simple Semantic-Based Data Storage Layout for Querying Point Clouds. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 72. [CrossRef]
125. Virtanen, J.P.; Daniel, S.; Turppa, T.; Zhu, L.; Julin, A.; Hyyppä, H.; Hyyppä, J. Interactive Dense Point Clouds in a Game Engine. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 375–389. [CrossRef]
126. Neuville, R.; Pouliot, J.; Poux, F.; Billen, R. 3D Viewpoint Management and Navigation in Urban Planning: Application to the Exploratory Phase. *Remote Sens.* **2019**, *11*, 236. [CrossRef]

127. Neuville, R.; Pouliot, J.; Billen, R. Identification of the Best 3D Viewpoint within the BIM Model: Application to Visual Tasks Related to Facility Management. *Buildings* **2019**, *9*, 167. [CrossRef]
128. Neuville, R.; Pouliot, J.; Poux, F.; de Rudder, L.; Billen, R. A Formalized 3D Geovisualization Illustrated to Selectivity Purpose of Virtual 3D City Model. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 194. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.