2026 European Conference on Computing in Construction
Corfu, Greece
July 12–15, 2026

AUTOMATED GENERATION OF DISASSEMBLY GRAPHS BASED ON DETAILED 2D SECTION DRAWINGS

Kasimir Forth[1], Mikhael Johanes[1], Elvire de Clermont Tonnerre[1], and Catherine De Wolf[1]
[1]ETH Zurich, Zurich, Switzerland

Abstract

Disassembly and circularity assessments are limited by the lack of structured, machine-readable data on material layers and connections, which are typically represented in expert-readable 2D section drawings. This paper explores how disassembly-relevant information can be automatically derived from such drawings represented as graphs using computer vision for segmentation, vision language models for material labeling, and language-based reasoning for semantic enrichment. The prototype extracts components and their connectivity and material information and generates graph representations capturing assembly hierarchies, enabling automated disassembly reasoning. This work marks a first step toward integrating heterogeneous data sources into unified graph models for circularity assessment.

Introduction

The built environment is one of the largest consumers of raw materials and a major contributor to global waste streams. Within Europe, construction and demolition activities generate close to 40% of all waste (García et al., 2024). Growing urbanization and rising living standards are expected to intensify this demand, particularly for mineral-based construction materials (OECD, 2019). Furthermore, as resource extraction and processing continue to drive greenhouse gas emissions, which are projected to rise to 4.6 Gt CO_2-equivalent annually by 2060 (UN Environment Programme, 2022), the construction sector faces increasing pressure to reduce its environmental footprint.

To address these challenges, the concept of a circular economy has gained prominence in the construction sector. It promotes keeping materials and components in the use stage for as long as possible by designing buildings that can be adapted, dismantled, dis- and reassembled rather than demolished. Achieving such reusability, however, depends on reliable knowledge of how building parts are connected and composed. This information of materials and components in existing contexts is often missing or inconsistently represented in current digital workflows, limiting the ability to evaluate a building's potential for disassembly.

Motivation and research gap

Existing circularity metrics, such as the material circularity indicator (MCI) (Goddin et al., 2019) and disassembly potential (Durmisevic, 2006), require detailed information on material composition, connection types, and assembly hierarchies. In previous studies (Forth and De Wolf, 2025), we showed that representing this disassembly information using knowledge graphs, hereinafter referred to as disassembly graphs, supports circular decision-making. However, the extraction of all relevant data for existing assemblies remains time consuming and dependent on expert knowledge.

Current digital methods for assessing disassembly potential rely on incomplete as-built data or manual modeling, while detailed two-dimensional (2D) section drawings contain rich latent information on assemblies and materials that remains underutilized. Automating the interpretation of such drawings can bridge the gap between design documentation and disassembly-oriented building information, enabling large-scale circularity assessments and disassembly planning and thus the reuse of components and materials.

This study investigates how disassembly graphs can be automatically derived from detailed 2D as-designed section drawings to represent the hierarchical and material relationships of existing building assemblies. We hypothesize that by combining vision-language reasoning techniques, it is possible to infer building component connectivity and semantic relationships sufficient to construct a graph representation that supports circularity assessments that focus on disassembly potential.

Background and related work

This section introduces the two main concepts of representing disassembly-related information using graphs and reasoning techniques for automatically extracting information from 2D architectural drawings.

Graph representations of disassembly information

The representation of disassembly information has evolved from manual heuristic assessments to sophisticated graph-based models that capture the complex interdependencies between building components. Fundamental to this domain is the work of Durmisevic, who established the theoretical groundwork for quantifying disassembly potential (Durmisevic, 2006). She proposed modeling building configurations as relational diagrams of components and their connections to represent physical dependencies. By analyzing these relational structures, Durmisevic demonstrated that the disassembly potential is inversely propor-

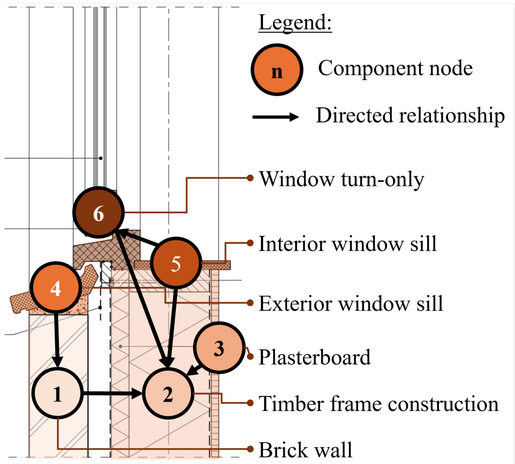

Figure 1: Example of information for disassembly potential of a wall and window according to van Vliet (2018)

tional to the complexity of the dependencies of components (Durmisevic, 2006). Specifically, hierarchical and parallel relational patterns facilitate higher recoverability compared to integrated, sequential dependencies.

Building on these topological foundations, Smith et al. introduced the disassembly sequence structure graph (DSSG), a formalism designed to optimize the disassembly process itself (Smith et al., 2012). Unlike static dependency diagrams, the DSSG models the process by generating a graph in which nodes represent specific parts and edges represent feasible removal sequences. This approach addresses the "selective disassembly" problem by allowing for the identification of optimal paths to remove specific target components for maintenance or reuse while minimizing the total number of operations.

In more recent applications focusing on circularity assessment, van Vliet operationalized Durmisevic's relational patterns to evaluate the circular potential of existing whole buildings (van Vliet, 2018). He simplified Durmisevic's approach by focusing on the most important properties, such as "type of connection", "assembly sequence", "accessibility to connection, as well as "assembly shape", "independency", "type of relational pattern" or "method of fabrication" to calculate a disassembly potential score. This method highlights the practical utility of graph representations in translating abstract circularity principles into quantifiable metrics for building elements.

Most recently, the focus has shifted toward using semantic knowledge graphs to manage the data required for these assessments. Bosma investigated the use of linked data to create digital product passports (DPPs) (Bosma, 2024). Instead of purely geometric or process graphs, Bosma utilizes an ontology-based graph structure, where products, materials, and their properties (such as location and warranty) are interlinked nodes. This semantic graph approach ensures that the disassembly information, such as material composition and manufacturer data, remains accessible and able to be queried.

Reasoning of 2D architectural drawings

In the domain of pixel-based analysis, convolutional neural networks (CNNs) and vision transformers (ViTs) have become the standard for tasks requiring visual pattern recognition, such as material detection and drawing classification. For instance, Pizarro et al. reviewed widespread methodologies involving CNNs for extracting walls, doors, and rooms from raster images. They note that these methods often discard essential topological metadata (Pizarro et al., 2022). More specialized applications include the work of Kairlapova et al., who successfully deployed CNN architectures like MobileNetV2 and ResNet50 to detect and classify building materials in 2D drawings to facilitate material passport generation (Kairlapova et al., 2025). Similarly, Carrara et al. compared ViTs against graph-based methods for classifying construction drawings into categories like discipline and project phase. They demonstrated that while pixel-based transformers are effective, they may fall short in capturing complex structural relations (Carrara et al., 2025a).
To address the limitations of raster approaches in capturing geometric and topological relationships, recent research has increasingly focused on vector and graph-based representations. Graph neural networks (GNNs) are gaining prominence for their ability to model drawings as connected entities rather than mere collections of pixels. Ko and Lee proposed a framework that converts architectural detail drawings into graph structures. Their framework utilizes GNNs to improve error detection and drawing classification by learning the interrelationships between drawing elements Ko and Lee (2025). Further advancing this domain, Carrara et al. introduced VectorGraphNet, a method that parses PDF drawings into scalable vector graphics graphs (Carrara et al., 2025b). By applying graph attention transformers, they achieved state-of-the-art results in line-level semantic segmentation, thus validating that structure-aware models can offer superior scalability and accuracy for complex technical drawings compared to conventional computer-vision-based techniques. Complementing these extraction methods, Schönfelder and König addressed the downstream challenge of data interoperability (Schönfelder and König, 2025). They introduced the drawing analysis ontology (DAnO), a framework designed to standardize the output of various computer vision tools into knowledge graphs, thereby enabling automated validation and semantic reasoning to refine the reconstructed building models.
Most approaches focus on segmentation and object detection on the floor plan level. To date, only Ko and Lee have focused on detailed architectural sections (Ko and Lee, 2025). However, they did not take textual input of legend and descriptions into account to further reason quantities, material, and connection information. Consequently, the work presented here extends the field toward multi-modal reasoning of detailed section drawings. It leverages segmentation, legend parsing, and graph-based modeling to infer assembly hierarchies and material relationships from

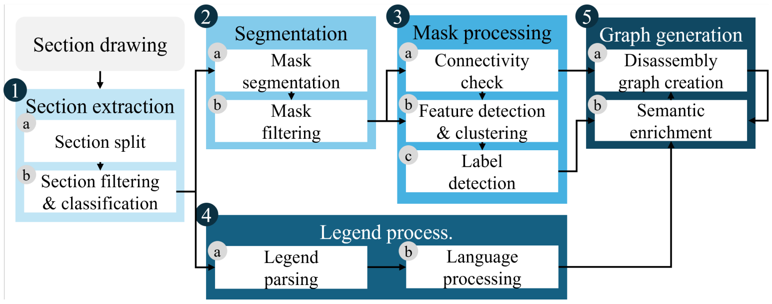

Figure 2: Overview of the proposed framework of deriving disassembly graphs based on 2D section drawings

as-designed documentation.

Method of generating disassembly graphs based on 2D section drawings

This study introduces a novel workflow for the automated generation of disassembly graphs based on 2D section drawings using computer vision, vision language models (VLM), and large language models (LLM) as well as graph-based reasoning techniques. Following Peffers' Design Science Research methodology Peffers et al. (2007), we identified the problem and defined the objective. Next, we designed and developed the overall framework, the steps of which are explained here. Finally, we demonstrated the framework with a case study and evaluated the results.

Overall framework

The overall framework consists of five main steps with several substeps, as shown in Figure 2. First, we split the whole drawing into its individual sections (1.a) and filter or classify them (1.b). Those section images that do not contain color gradients or legend text are further processed for segmentation (2). In the second step, each section is segmented into masks (2.a), and the masks are filtered (2.b): meaningful masks are kept, and almost blank masks are discarded. In the next step (3), the filtered masks from the second step are further processed and combined with the processed legend information. In the first substep (3.a), the connectivity between neighboring masks is identified. Next, features are detected (3.b), and finally, every mask is labeled either according to the detected features or the identified legend association (3.c). The section that was classified as legend is further processed by parsing and correcting the text (4.a) and then restructuring the text according to geometric and semantic information (4.b). In the final step (5), the disassembly graph is generated by, first, creating nodes for each mask and edges for each connection (5.a) and then enriching the results of the labeling and legend information (5.b). Depending on the drawing style, the order of these steps may change, but the general steps can be robustly applied to several drawing styles.

Section extraction

The section extraction-and-filtering pipeline automates the decomposition of sections across the entire drawing. Initial regions of interest are isolated by detecting structural contours within the image, which separates potential layout blocks from the background. Text-only areas are excluded using CLIP-based semantic filtering (Radford et al., 2021). These candidates are then validated using a visual classifier that checks for low color saturation and high edge density. This process effectively distinguishes valid technical sections, which are characterized by dense, monochromatic line work, from non-geometric artifacts, such as logos or photorealistic renderings.

Segmentation

We apply an automatic mask generator based on a Segment Anything Model (SAM) (Kirillov et al., 2023) that densely samples point prompts across the drawing to produce a large set of overlapping candidate segments. These proposals are subsequently post-processed to resolve overlaps and extract a set of disjoint graphical regions corresponding to distinct drawing components. To ensure reproducibility and stabilize color-encoded outputs, each detected mask is assigned a fixed, distinct color generated via a deterministic color scheme. Each cropped mask is evaluated using grayscale statistics, including brightness distribution, proportion of dark pixels, edge variance, and intensity histogram entropy. Segments are retained only if they exhibit (i) sufficient dark content relative to the background, (ii) meaningful edge structure indicative of technical line work or texture, and (iii) adequate intensity diversity throughout the region. Near white or homogeneous fragments that do not meet these criteria are discarded.

Mask processing

Different processing steps are necessary to bridge the gap between the segmented masks and the processed legend in order to generate semantic-rich disassembly graphs. We propose three main processing steps: connectivity check (4.a), feature detection and clustering (4.b), and label detection (4.c).

The first step checks the connectivity between masks representing physical adjacency between them. In technical drawings, lines of masks that should theoretically touch are often separated by tiny gaps of 1–2 pixels due to rasterization (converting vector lines to pixels) or anti-aliasing (smoothing edges). To take this into account, we use a 3x3 kernel dilation, which expands every segment by 3 pixels in all directions. Now the artificially expanded segment bridges tiny gaps, ensuring that segments which are almost touching are recorded as adjacent. This robust adjacency detection ensures that compound structures (e.g., a wall composed of multiple material layers) are correctly linked as a coherent assembly.

To enable classification of similar materials within each mask, the system extracts a high-dimensional feature vector for each segment. While models like CLIP are powerful for semantic image-text alignment (Radford et al., 2021), we used DINO (self-supervised Distillation with No labels) in our implementation (Siméoni et al., 2025). The decision to use DINO stems from the nature of technical drawings. Architectural component materials are defined by local geometric textures (hatching patterns) rather than the global semantic contexts. The extracted features are subsequently clustered using K-Means algorithm, which groups segments with visually similar hatching patterns into common classes. This visual clustering serves as a fallback mechanism. If a specific segment lacks a direct text label from the next step, then its material properties can be inferred from its cluster neighbors that have been successfully identified.

The final step bridges the gap between the component segment connectivity graph and the parsed legend data. While the previous steps characterize the shape of the segments where they touch, this step identifies what label it contains by detecting numerical callouts. The system employs a VLM, specifically configured with a structured prompting strategy, to scan the original section images and the segments. The model is provided with the cropped section and the bounding box coordinates of the segments. To further facilitate spatial reasoning, the input image is augmented with a visible grid aligned with the image's pixel resolution. This grid provides an explicit, shared spatial reference frame that simplifies the localization and description of leader lines and numerical callouts by the language model. It is then tasked with identifying leader lines and numerical indicators pointing to specific geometries. The output is a structured mapping linking a segment ID to a legend number. This direct assignment allows the system to propagate the rich material metadata extracted from the legend in the next step to the corresponding nodes in the graph, thereby completing the semantic enrichment of the graph in the final step.

Legend parsing

To bridge the gap between abstract geometric segments and their physical properties, we detect and interpret the drawing's legend. Initial attempts to digitize this information relied on conventional optical character recognition (OCR) engines, such as Tesseract. However, these methods proved insufficient for the provided dataset. The low resolution and rasterization artifacts inherent in the technical drawings led to significant character misrecognition, resulting in incoherent text outputs that lacked semantic continuity.

To overcome these limitations, we use a VLM, in our case OpenAI's "GPT-4o-mini" (OpenAI et al., 2024). Unlike standard OCR, which processes characters individually, the VLM analyzes the visual context of the legend table, allowing it to infer meaning even from pixelated text. The model is prompted to extract the legend text and restructure it into a machine-readable dictionary, as shown for one layer in 4.b in Figure 3. This prompting strategy explicitly instructs the model to decompose the raw text descriptions into granular attributes for each layer. The resulting data structure differentiates for every component between material composition and layer thickness and, if available, connection type. We have, to date, implemented this workflow for only one style of legend and need to adapt it to further styles in future.

Graph generation

The final stage of the framework synthesizes the geometric, topological, and semantic data into a unified graph database. This step transforms the isolated segmentation masks into a disassembly graph that can be queried, where the building assembly is represented as a network of interconnected components. We use labeled property graphs since component properties can directly be stored in nodes and connection properties in edges, and implement the disassembly graph using Neo4j.

For every segmentation mask, a distinct component node is instantiated within the graph database. To ensure full traceability and visual consistency between the disassembly graph and the original drawing, each node is populated with a comprehensive set of properties, including a unique segment identifier, the source case study and section reference, and the specific RGB color values of the mask, which allow for manual traceability. Additionally, classification data such as the detected label and the cluster ID from feature analysis are stored as node properties to serve as keys for subsequent semantic mapping.

Following node creation, the physical relationships between components are encoded as edges to model the structural topology of the assembly. Using the adjacency list generated during the connectivity analysis, the system establishes connection relationships between interacting nodes, effectively distinguishing between isolated elements and composite structures.

Finally, the graph is enriched with the detailed metadata parsed from the legend. By querying the label property of each node against the structured legend dictionary, the system infers all available information, such as material composition and layer thickness, into the geometric entities.

Case study and results

To validate the proposed framework, we prototypically implemented the workflow and evaluated the segmentation, connectivity, and label detection step in two case studies to show the accuracy and stability performance at each step.

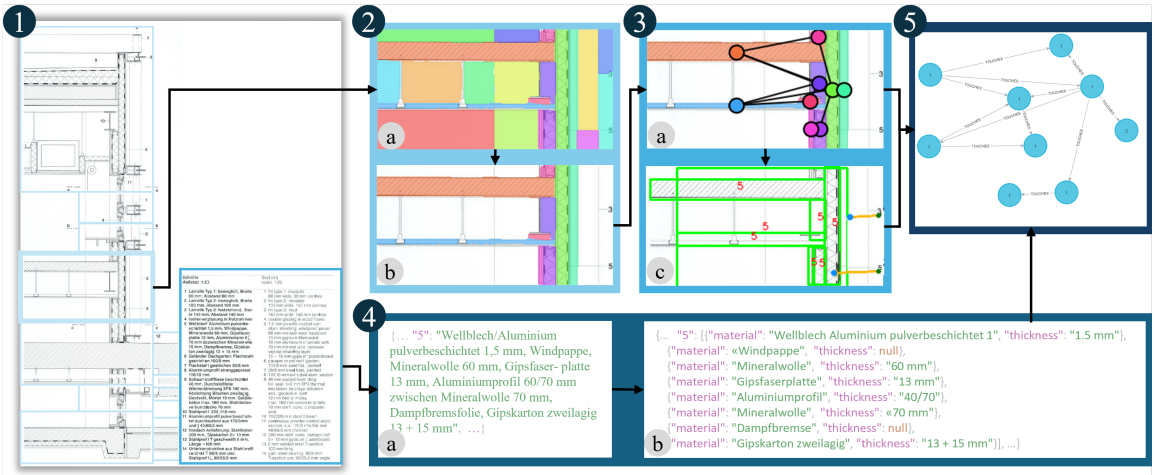

Figure 3: Overview of the results of one case study (CS 2) from DETAIL Magazin (2019/04, p.37, based on b720 Arquitectos), highlighting the section split (1), the segmentation (2.a) and its filtering (2.b) of one selected section, the connectivity check (3.a) and detected labels (3.c), the related legend excerpt (4), and the resulting disassembly graph of this section (5).

Case studies

We selected two exemplary detailed section pages from the architecture trade magazine *DETAIL*, one containing six sections and the second four sections. These reflect a standardized style of detailed section drawings and contain a legend in German. Figure 3 shows one exemplary section with the results of each step. The case studies show multi-layered facades, challenging the segmentation, connectivity, and labeling algorithm to distinguish repetitive geometries, continuous layers, and air gaps.

Results and discussion

In the following subsections, we evaluate three main steps of the overall workflow, focusing on the segmentation, the resulting connectivities, and the identification of labels for the relevant segments. We ran the implementation five times for both case studies to quantitatively validate accuracy and stability for the three selected steps, as well as analyzing error patterns qualitatively selecting one example section.

Segmentation

The segmentation performance was evaluated by comparing predicted instance masks against manually annotated ground truth across all ten section drawings from both case studies. Instance-level evaluation was performed using greedy matching at an IoU threshold of 0.5, resulting in a precision of 52.4%, a recall of 61.5%, and an F1-score of 56.6%. These results indicate limited agreement at the instance level.

Further analysis shows that the instance-level scores are primarily affected by systematic over- and under-segmentation rather than poor spatial localization. When instance boundaries are ignored and evaluation is performed on pixel-wise overlap between the unions of ground truth and predicted masks, the measured agreement is substantially higher, with a precision of 73.7%, a recall of 93.2%, and an F1-score of 82.3%. This discrepancy indicates that most relevant regions are spatially detected but not consistently partitioned into the correct instances.

Qualitative inspection, shown in Figure 4, confirms this behavior. The overlay view (top) shows good geometric alignment between predictions (colored fill) and ground truth (gray outlines), indicating that errors are primarily due to incorrect mask extent rather than mislocalization. The middle view highlights under-segmentation, where multiple adjacent components are merged into a single predicted instance, producing overly large masks that introduce false-positive regions. The bottom view illustrates over-segmentation, where elements with large hatched regions are split into multiple predicted instances, particularly across internal boundaries introduced by hatch patterns.

These results highlight the need for more domain-specific instance segmentation approaches for architectural and construction drawings. Although the use of SAM-based segmentation is promising, the model is not well adapted to the abstraction level and graphical conventions of section drawings, leading to systematic under- and over-segmentation. In particular, thin, contiguous, and hierarchically composed elements are not consistently grouped into coherent instances despite being geometrically well localized. This limitation suggests that generic image-driven segmentation models require additional architectural priors, drawing-aware constraints, or task-specific fine-tuning to reliably support instance-level reasoning in this domain.

Connectivity results

The performance of the connectivity algorithm was quantitatively assessed by comparing the automatically generated connectivity graphs against a manually annotated ground truth across ten detailed 2D architectural sections. Missing connections lead to false negatives (red), and wrong connections lead to false positives (orange), as

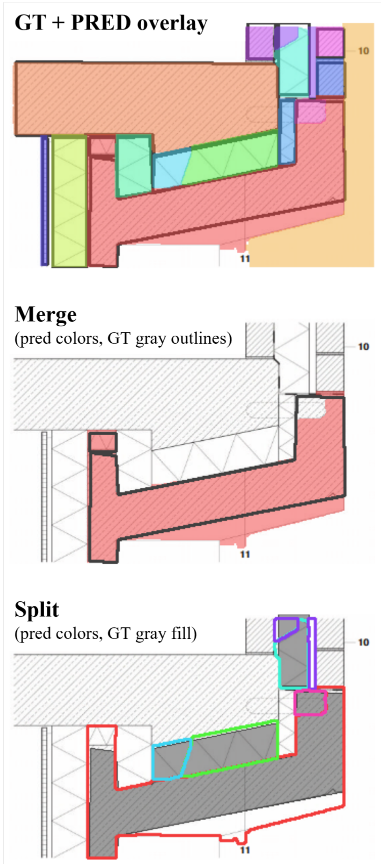

Figure 4: A qualitative comparison of ground truth and predicted segmentation for a representative section drawing shows generally sufficient alignment but inconsistent instance boundaries due to over- and under-segmentation.

shown exemplarily in Figure 5. The system achieved balanced and stable performance in all five iterations, yielding a precision of 82.52%, a recall of 87.41%, and a resulting F1 score of 84.89%. These metrics indicate that the framework is highly reliable for identifying standard physical adjacencies, effectively translating segmented mask data into a functional connectivity graph. The near parity between precision and recall suggests that the model does not suffer from a systemic bias toward over-prediction or under-prediction, providing a stable baseline for automated disassembly reasoning.

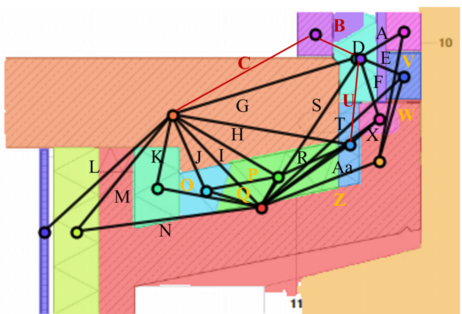

Figure 5: Evaluation of the results of the automatic connectivity check based on previous segmentations of one section example

A critical challenge identified during the evaluation is missed connections due to insufficient proximity between masks. In several instances, as illustrated in Figure 5, the human expert identified essential structural connections that the algorithm failed to detect because the masks were separated by small gaps. These gaps, often only a few pixels wide, frequently result from anti-aliasing artifacts during segmentation or the presence of thin architectural membranes that do not register as overlapping pixels. Because the current logic relies on strict pixel-to-pixel contact (extending by 3 pixels), these semantically connected components are treated as isolated entities. This limitation is particularly evident in complex joints where multiple materials converge but do not perfectly intersect in the digital representation.

Furthermore, the accuracy of the connectivity graph is heavily dependent on the upstream success of the computer vision segmentation phase. As highlighted in Figure 5, the absence of specific material masks directly results in a cascade of missing connections. These missing segments or over-segmented masks, such as unidentified insulation layers (light blue and green) in Figure 5, represent most of the recorded false positive connections. When the segmentation model fails to identify a mask for a specific material layer, the graph generator cannot establish any adjacency for that component, leading to incomplete assembly hierarchies. This underscores that the robustness of the final disassembly graph is intrinsically linked to the recall of the initial segmentation and labeling processes.

Segment label identification

Table 1 shows the average results of precision, recall, and F1 score as well as their standard deviations (σ) of all five iterations for each case study and overall (mean).

Table 1: Performance metrics for label identification across five iterations and the overall average

Average	**Precision**	**Recall**	**F1 Score**
CS 1	50.47%	86.00%	61.04%
CS 2	27.83%	100.00%	43.52%
Mean	37.68%	90.75%	52.91%
Std. Dev.	**Precision**	**Recall**	**F1 Score**
CS 1	9.58%	21.91%	2.39%
CS 2	1.82%	0.00%	2.23%
Mean	1.41%	14.58%	2.37%

The results show that case study 1 achieved a balanced performance, with an average F1 score of 61.04%, characterized by a recall of 86.00% and precision of 50.47%. However, it exhibited significant instability in detection rates. This was indicated by a high standard deviation in recall (σ = 21.91%). Conversely, case study 2 demonstrated perfect retrieval stability (recall 100.00%, $\sigma = 0.00\%$) but significantly lower precision (27.83%), resulting in a lower F1 score of 43.52%. This comparison highlights that while the system reliably detects all labels in complex configurations (CS 2), it is prone to assign labels too broadly compared to the more balanced but variable results of standard details (CS 1).

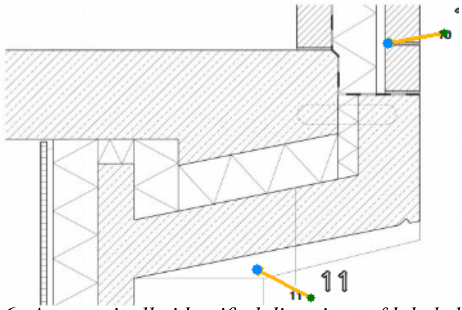

Figure 6: Automatically identified directions of labels leading to sections of one ion example.

The sources of inaccuracy were further investigated by visually inspecting and describing one exemplary section using the intermediate and final outputs. Figure 6 illustrates the system's interpretation of the leading lines. The green start point identifies the numeric label text, indicated by a green start point, and the blue end point the identified end point at the target segment, connected by a yellow leader line. While the leading line of label "10" accurately follows the initial direction, the leading line of label "11" deviates significantly, causing wrong labeling.

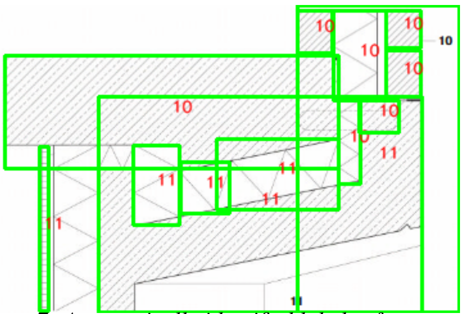

Figure 7: Automatically identified labels of one section example.

Figure 7 presents the resulting label assignments, revealing distinct success and failure modes. The system demonstrates robust performance in standard conditions, as evidenced by the correct labeling of the four top right segments. However, errors primarily arise from geometric inconsistencies in vector interpretation. A notable example occurs in the bottom-left region, where a segment is incorrectly labeled as "11" because the interpreted leading line vector deviates from its intended termination point and intersects a neighboring mask, as shown in Figure 6 . Additionally, the system exhibits a tendency to assign false positives in ambiguous zones. Two empty segments are incorrectly assigned label "10" despite the absence of explicit vector connectivity, suggesting an over-reliance on spatial proximity when graphical signals are weak.

In summary, the results indicate that while the VLM-based extraction is highly effective at retrieving semantic data (high recall), the geometric mapping mechanism requires refinement. The prevalence of false positives highlights the need for stricter vector constraints and collision detection algorithms to resolve ambiguities in dense or complex drawing regions. Thereby, we are aligning the module's precision with its high retrieval capability. In practice, better performance is observed in sections with larger components, while smaller or finer elements tend to generate more ambiguous mappings.

Limitations

The primary limitation pertains to the validated case studies. The current workflow is optimized using high-quality, standardized architectural detail drawings from a DETAIL magazine. Consequently, the system's robustness against the heterogeneity of broader architectural drawings, such as archival drawings, varying hatching standards, or hand-drawn sketches, remains unverified. The pipeline's adaptability to unstructured annotations or non-standardized material indications has not yet been evaluated.

Methodologically, quantitative benchmarking was restricted to the geometric processing modules. More specifically, it confirm only the accuracy of segmentation, connectivity detection, and label identification. Performance of legend processing and visual feature extraction have not yet been validated. Finally, the resulting disassembly potential calculations for every section have yet to be validated against a manual ground truth.

Technically, the prototype utilizes a specific configuration of models (SAM, DINOv3, GPT-4o) without comparative analysis. Ablation studies benchmarking these choices against alternative architectures, e.g., CLIP for feature extraction or open-source LLMs for parsing, were not conducted. Therefore, the current implementation may not represent the optimal balance between computational cost and inference accuracy.

Conclusion and outlook

This study proposes a novel computational framework for the automated generation of knowledge graphs interpreting disassembly-related data from 2D architectural section drawings. By integrating geometric segmentation (SAM), visual feature encoding (DINOv3), and semantic parsing (VLM), the pipeline successfully converts detailed section drawings with dense domain knowledge into disassembly graphs that can be queried. This transformation bridges a critical gap in the built environment: It enables the systematic assessment of reusability of components and materials and assembly hierarchies from legacy data sources that were previously inaccessible to digital circularity tools.

Future research will focus on validating the end-to-end fidelity of the generated disassembly graphs by comparing disassembly potential calculation results against manually curated ground truth. Next, the pipeline's robustness will be tested on heterogeneous datasets that extend beyond standardized details to include diverse archival and hand-drawn documentation. Additionally, systematic ablation studies will benchmark alternative architectures. We will compare different models for computer vision, VLM, and LLM to optimize the trade-off between computational cost and inference accuracy, and focus on connection type properties. Finally, we aim to research how to link and fuse data with other data sources such as point clouds and BIM models (Forth and De Wolf, 2025) to enable a data-driven workflow for holistic circularity assessments.

Acknowledgments

This research has been funded by the SWIRCULAR project supported by Innosuisse (107.235 FS-EE). A collaborating author was supported through a Design++ fellowship program at ETH Zurich.

References

Bosma, J. A. M. (2024). From Material Passports to Digital Product Passports Creating and validating Linked Data-Based Digital Product Passports for the AEC industry.

Carrara, A., Nousias, S., and Borrmann, A. (2025a). Content-based classification of construction drawings : a comparative study of vision transformers and graph attention networks. In EG-ICE 2025: International Workshop on Intelligent Computing in Engineering. University of Strathclyde Publishing, Edinburgh, UK.

Carrara, A., Nousias, S., and Borrmann, A. (2025b). VectorGraphNET: Graph Attention Networks for Accurate Segmentation of Complex Technical Drawings. Journal of Computing in Civil Engineering, 39(6):04025085.

Durmisevic, E. (2006). Transformable building structures. Design for dissassembly as a way to introduce sustainable engineering to building design & construction. [s.n.], [S.l.].

Forth, K. and De Wolf, C. (2025). Formalizing information for disassembly potenttial of buildings using bim and labeled property graphs. In Proceedings of the 2025 European Conference on Computing in Construction, volume 6, Porto, Portugal. EC3.

García, C., Caro, J., Gallo, G., and Tonini, F. (2024). Techno-economic and environmental assessment of construction and demolition waste management in the European Union - Status quo and prospective potential. Technical report, European Commission.

Goddin, J., Marshall, K., Pereira, A., Tuppen, C., Herrmann, S., Jones, S., Krieger, T., and Lenges, C. e. (2019). Circularity Indicators: An Approach to Measuring Circularity.

Kairlapova, A., Forth, K., Carrara, A., and Borrmann, A. (2025). Material detection and classification in 2d architectural drawings using computer vision for material passports. In Proceedings of the 2025 European Conference on Computing in Construction, volume 6, Porto, Portugal. EC3.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment Anything. arXiv:2304.02643.

Ko, J. and Lee, D. (2025). Graph neural networks for classification and error detection in 2D architectural detail drawings. Automation in Construction, 170:105936.

OECD (2019). Global Material Resources Outlook to 2060: Economic Drivers and Environmental Consequences. Technical report, OECD Publishing, Paris. ISBN: 9789264307445.

OpenAI, Hurst, A., Lerer, A., and Goucher, A. P. e. (2024). GPT-4o System Card. arXiv:2410.21276 [cs].

Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2007). A design science research methodology for information systems research. Journal of Management Information Systems, 24(3):45–77.

Pizarro, P. N., Hitschfeld, N., Sipiran, I., and Saavedra, J. M. (2022). Automatic floor plan analysis and recognition. Automation in Construction, 140:104348.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision.

Schönfelder, P. and König, M. (2025). Ontology-based reasoning in automatic floor plan analysis. Advanced Engineering Informatics, 68:103761.

Siméoni, O., Vo, H. V., Seitzer, M., Baldassarre, F., and et al. (2025). DINOv3. arXiv:2508.10104 [cs].

Smith, S., Smith, G., and Chen, W. H. (2012). Disassembly sequence structure graphs: An optimal approach for multiple-target selective disassembly sequence planning. Advanced Engineering Informatics, 26(2):306–316.

UN Environment Programme (2022). 2022 Global Status Report for Buildings and Construction: Towards a Zero-emission, Efficient and Resilient Buildings and Construction Sector.

van Vliet, M. (2018). Disassembling the steps towards Building Circularity Redeveloping the Building Disassembly assessment method in the Building Circularity Indicator. Master's thesis, Eindhoven University of Technology.