HIERARCHICAL POINT–IMAGE FUSION WITH CONTEXT ENHANCEMENT FOR PANOPTIC SYMBOL SPOTTING IN CAD DRAWINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Panoptic symbol spotting in computer-aided design (CAD) drawings jointly identifies countable object instances and assigns semantic labels to all graphical primitives, including those representing uncountable stuff. Reliable recognition depends on both the geometry of individual primitives and the spatial arrangement of surrounding strokes, with these cues encoded differently in vector and raster representations. We propose a point–image fusion framework that combines bidirectional backbone interaction with local context enhancement before primitive fusion, bringing complementary visual context overlooked by vector-only methods into primitive representation learning. Specifically, we introduce Hierarchical Point–Image Fusion (HPIF) to enable spatially aligned exchange of local geometric and visual cues, enriching point features with multiscale visual context from the image branch and image features with geometric information from vector primitives. To strengthen each primitive's representation with surrounding geometric and visual evidence, our Primitive-Oriented Context Enhancement (POCE) module combines radius-bounded attention over same-layer primitives with regular and shifted window attention on image features, further improving panoptic recognition. Extensive experiments demonstrate that our framework outperforms previous methods on FloorPlanCAD and ArchCAD-Public, achieving state-of-the-art panoptic quality (PQ) of 92.7 and 90.6, respectively. Code is available at `http://anonymous.4open.science/r/code-BED2`.

1 INTRODUCTION

Panoptic symbol spotting converts CAD drawings into semantic and instance annotations at the level of graphical primitives. Given a drawing composed of lines, arcs, circles, and other geometric entities, a model must recognize both countable objects, such as doors and furniture, and uncountable structures, such as walls. FloorPlanCAD (Fan et al., 2021) established this task as a joint problem of semantic labeling and instance recognition. These primitive-level annotations provide a basis for structured drawing interpretation and building information extraction. ArchCAD (Luo et al., 2025) further extends evaluation to a broader range of architectural drawings and symbol categories. A central difficulty is that an individual primitive rarely describes a complete object: similar line segments can belong to different symbols, and recognizing their roles requires understanding how neighboring strokes form a larger structure. The task therefore calls for representations that retain primitive geometry while making contextual evidence available to each prediction.

Progress in CAD symbol spotting has been closely tied to how graphical primitives are represented. Existing methods model primitive relationships through graphs (Zheng et al., 2022) or encode geometry using points and sampled line segments (Liu et al., 2024a; Wei et al., 2025). Point- and line-based representations preserve the geometry of individual primitives, but do not explicitly encode the dense visual context of rendered drawings. Image features capture surrounding stroke patterns and spatial configurations that complement primitive geometry. Combining these representations provides richer evidence for recognizing symbols and identifying their constituent primitives.

To exploit this complementarity, geometric and visual information should interact during feature extraction as well as at the decoding interface. Multiscale image features can supply point represen-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tations with the surrounding stroke context, while point features can provide image representations with the geometry of individual primitives. DPSS (Luo et al., 2025) combines separately encoded geometric and visual features before decoding, demonstrating the value of the two representations, but its backbones build intermediate features independently. Exchanging spatially aligned features across backbone stages would allow visual context and primitive geometry to guide subsequent feature transformations in both branches. The resulting representations can further benefit from the local arrangement of nearby primitives and surrounding image structures. Together, these considerations motivate backbone interaction followed by local context enhancement for panoptic prediction.

We propose a point–image fusion framework that integrates hierarchical feature interaction with local context enhancement for CAD panoptic symbol spotting. After sampling, the point branch uses Point Transformer V3 (PTv3) (Wu et al., 2024) to encode primitive geometry, while the image branch uses UHRNet to extract dense multiscale features from the rendered drawing. We design Hierarchical Point–Image Fusion (HPIF) to exchange spatially aligned features between irregular CAD point tokens and dense image grids across backbone stages. Using shared CAD coordinates, HPIF samples image features at point locations and projects point features onto image grids, allowing geometry and appearance to guide feature learning in both branches. At the backbone outputs, Primitive-Oriented Context Enhancement (POCE) aggregates local geometric and visual context to strengthen primitive representations. Image features are then read along constituent segments and pooled to the primitive level. The resulting image and point features are fused and passed to the CAD decoder. Experiments on FloorPlanCAD and ArchCAD-Public demonstrate that our framework achieves state-of-the-art panoptic quality (PQ) of 92.7 and 90.6, respectively.

Our main contributions are threefold:

1. We construct a hybrid point–image fusion framework that combines geometric features of sampled CAD primitives with dense multiscale image features for panoptic symbol spotting.
2. We design HPIF to align and exchange point and image features across backbone stages, transferring visual features of surrounding stroke patterns to primitive representations and primitive geometric cues to spatially corresponding image locations.
3. We introduce POCE, which combines radius-bounded, same-layer primitive attention with image window attention, using neighborhoods appropriate to each representation.

2 Related Work

2.1 CAD Panoptic Symbol Spotting

Earlier work studied architectural symbol spotting (Rezvanifar et al., 2019; 2020). Building on panoptic segmentation (Kirillov et al., 2019), Fan et al. (2021) introduce FloorPlanCAD and formulate panoptic symbol spotting as the joint recognition of countable thing instances and semantic labeling of uncountable stuff. Their PanCADNet combines Faster R-CNN (Ren et al., 2015) for instance detection with graph convolutions (Kipf & Welling, 2017) for primitive semantic labeling. Fan et al. (2022) propose CADTransformer, which constructs primitive tokens from convolutional features sampled at primitive centers in rendered drawings. GAT-CADNet encodes vector geometry through graph attention and primitive adjacency prediction (Zheng et al., 2022). Graph representations also support floorplan segmentation in VectorFloorSeg (Yang et al., 2023). The SymPoint family represents primitives as points, with attention that incorporates primitive connections in SymPoint and layer features and position guidance during training in SymPoint V2 (Liu et al., 2024a;b). To capture geometry within each primitive, CADSpotting aggregates densely sampled point features, whereas VecFormer encodes connected line segments (Yang et al., 2026; Wei et al., 2025). The point- and line-based methods preserve primitive geometry without explicitly encoding dense raster context. TextCAD incorporates semantic cues from native textual annotations into primitive representations, but its reliance on these inputs limits direct application to drawings without textual annotations (Gong et al., 2026). DPSS combines visual and geometric information by encoding images and primitives in separate branches and adaptively fusing image features sampled at primitive centers with geometric features before decoding (Luo et al., 2025). However, its two encoders remain separate during feature extraction, leaving primitive geometry and raster appearance unable to

guide each other's intermediate representations. Our framework instead enables bidirectional interaction throughout the backbone and aggregates image features along constituent segments, achieving improved panoptic recognition.

2.2 Hybrid Backbones and Feature Interaction

Hybrid architectures combine the spatial inductive biases of convolution with the capacity of Transformers to model relationships across image regions, enabling local structure and broader context to interact during feature extraction. Conformer (Peng et al., 2021) couples CNN and Transformer branches, while ViT-Adapter (Chen et al., 2023) injects convolutional priors into a plain ViT (Dosovitskiy et al., 2021) for dense prediction. ViT-CoMer (Xia et al., 2024) further enables bidirectional interaction between convolutional multiscale features and Transformer features. For point–image learning, 4D-Net (Piergiovanni et al., 2021) aligns image and point-cloud features through geometric projection and learned connections, and DeepInteraction (Yang et al., 2022) couples LiDAR and image representations through bidirectional feature interaction. Our setting couples CAD primitive geometry with rasterizations of the same drawing, requiring spatially aligned exchange between irregular tokens and dense image grids.

3 Method

Overview. In this section, we present our point–image fusion framework, as illustrated in Figure 1. We introduce the point–image backbone and HPIF for exchanging geometric and visual features during feature extraction. We then describe POCE for local context enhancement. The refined features are then fused for each primitive and decoded into semantic labels and instance masks to form the panoptic prediction.

3.1 Point–Image Backbone

Given a CAD drawing with primitives $\mathcal{P} = \{p_i\}_{i=1}^{N}$ and its raster rendering I, panoptic symbol spotting predicts a semantic category for every primitive and groups primitives belonging to each countable object instance. The category set contains countable things and uncountable stuff.

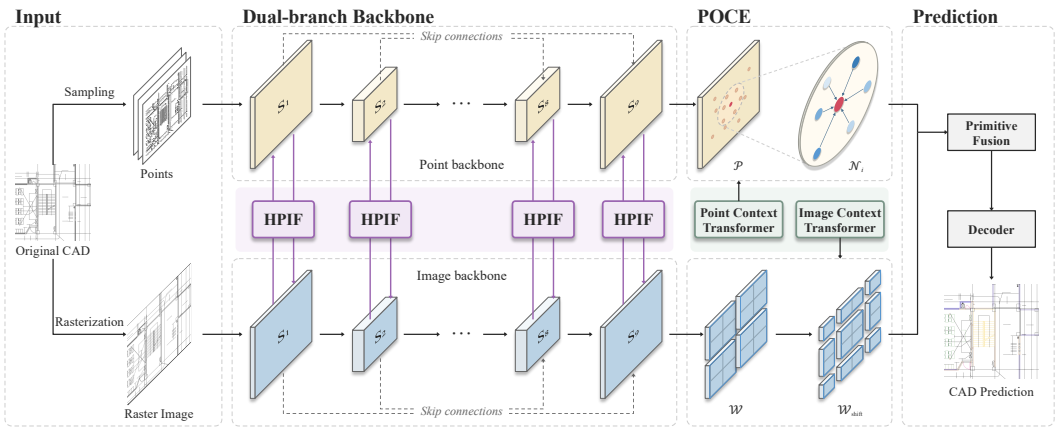
We choose PTv3 for its efficient modeling of spatial relationships and its ability to scale attention to large receptive fields. Through line sampling, we represent each primitive as connected segments, each encoded by a midpoint token with geometric attributes (Wei et al., 2025). Token coordinates are (x, y, z), where (x, y) is the normalized midpoint and z is the normalized CAD layer identifier when layer priors are used. Otherwise, $z = 0$ and all primitives share one layer. PTv3 extracts multiresolution features through attention within serialized patches, varying serialization orders across blocks to connect different neighborhoods (Wu et al., 2024).

CADTransformer and DPSS use HRNet to extract image features (Fan et al., 2022; Luo et al., 2025; Wang et al., 2021). We choose UHRNet because it reallocates computation toward lower resolutions and progressively propagates semantic features back to higher resolutions (Wang et al., 2022). We render CAD drawings in black and white to obtain I. UHRNet produces multiresolution image features for HPIF interaction during downsampling and resolution recovery.

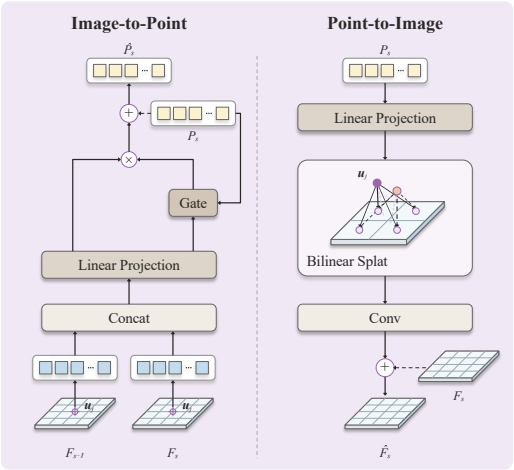
3.2 Hierarchical Point–Image Fusion

HPIF enables geometric and visual representations to guide feature extraction in both backbones. It contains an image-to-point (I2P) path and a point-to-image (P2I) path. I2P supplies point tokens with convolutional appearance features and multiscale spatial information, while P2I injects the geometry encoded by point tokens into spatially corresponding image grids. Shared CAD coordinates connect the irregular point tokens and dense image grids, allowing these complementary features to enter subsequent transformations in both branches.

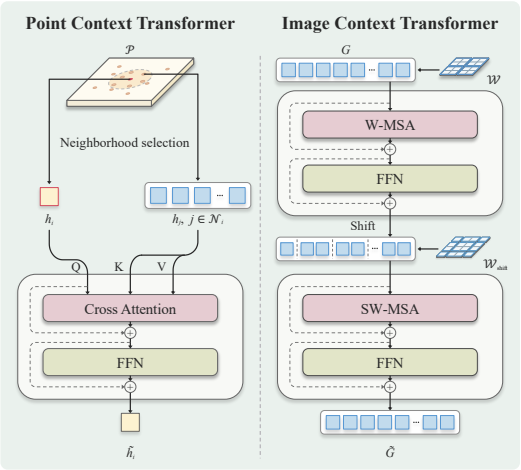
The I2P path enriches the geometric point representation with visual features extracted from the raster drawing. Let $\mathbf{x}_j$ denote a point token feature at normalized drawing location $\mathbf{u}_j \in \mathbb{R}^2$. We use image feature maps F_{s-1} and F_s from the previous and current stages, respectively. We sample both maps at $\mathbf{u}_j$ and fuse their concatenated features through a gated residual update:



(a) Overall architecture



(b) HPIF



(c) POCE

Figure 1: **Overview of the proposed framework.** (a) The point and image backbones exchange features through HPIF, followed by POCE, primitive fusion, and decoding. (b) HPIF aligns image and point features through gated image-to-point (I2P) fusion and point-to-image (P2I) projection. (c) POCE uses Point and Image Context Transformers to aggregate neighboring primitive features and image window features, respectively.

$$\begin{aligned} \mathbf{z}_j &= \phi(\mathrm{Concat}[\mathcal{B}(F_{s-1}, \mathbf{u}_j),\, \mathcal{B}(F_s, \mathbf{u}_j)])\,, \\ \widehat{\mathbf{x}}_j &= \mathbf{x}_j + \alpha\, g(\mathbf{x}_j, \mathbf{z}_j) \odot \mathbf{z}_j, \end{aligned} \tag{1}$$

where $\mathcal{B}$ denotes bilinear sampling, ϕ is a linear projection to the point channels, g is a channel gate, and α is a learned residual scale.

The P2I path makes the geometry encoded by point tokens available to the image branch, so subsequent convolutions can use both raster appearance and vector geometry. For either image map $F \in \{F_{s-1}, F_s\}$, we project point features $\mathbf{X}$ to its channels and distribute them onto its grid using drawing coordinates $\mathbf{U}$:

$$\begin{aligned} D &= \mathrm{Splat}(W\mathbf{X}, \mathbf{U}), \\ \widehat{F} &= F + \beta\ \mathrm{Conv}_{1\times1}(D), \end{aligned} \tag{2}$$

where W is a linear projection and β is a learned residual scale. Splat distributes features to the four adjacent grid locations with bilinear weights and normalizes accumulated features by the corresponding weight sums.

Both paths use the same CAD coordinates to preserve spatial correspondence. We apply HPIF at four downsampling stages and four stages of resolution recovery, allowing subsequent feature extraction in each branch to incorporate information from the other.

3.3 PRIMITIVE-ORIENTED CONTEXT ENHANCEMENT

POCE equips each primitive with information from nearby drawing elements for panoptic prediction. Building on the representations produced by HPIF, the module combines a Point Context Transformer (PCT) operating on primitive features with an Image Context Transformer (ICT) operating on the aggregated image grid. PCT captures relationships among nearby primitives within the same CAD layer. ICT incorporates surrounding visual information into the features that will be read along each primitive. Together, the two modules enrich primitive representations with geometric and visual context for subsequent panoptic decoding.

PCT uses the local arrangement of primitives to enrich the feature of each prediction unit. Let $\mathbf{h}_i$ denote the feature pooled from the constituent segments of primitive p_i, and let $\bar{\mathbf{c}}_i$ be the mean of its constituent segments' XY midpoints. For neighborhood selection, we use normalized drawing coordinates to identify the K nearest other primitives in the same layer, denoted by $\mathcal{K}_i$. Restricting these candidates to XY radius r gives the spatial neighborhood

$$\mathcal{N}_i = \{j \in \mathcal{K}_i : \|\bar{\mathbf{c}}_j - \bar{\mathbf{c}}_i\|_2 \le r\}.$$

For each attention head, $\mathbf{q}_i$ is projected from layer-normalized $\mathbf{h}_i$, and $\mathbf{k}_j$ and $\mathbf{v}_j$ from layer-normalized neighbor features $\mathbf{h}_j$. The resulting context is

$$\mathbf{m}_i = \sum_{j \in \mathcal{N}_i} \mathrm{softmax}_j\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d}} + b_{ij}\right)\mathbf{v}_j, \tag{3}$$

where d is the head dimension and b_{ij} is computed by an MLP from the relative XY displacement and distance, normalized by r. The contexts from all heads are projected into a residual update, followed by a residual feed-forward block that produces $\widetilde{\mathbf{h}}_i$.

ICT uses window attention to enrich features read along a primitive with information from the surrounding image region. Multiscale image outputs are channel-pooled, resized, and concatenated into a grid G. After a channel projection, we represent the grid as cell features $\{\mathbf{f}_t\}$ and group them by spatial location into nonoverlapping windows. For a window with cell indices $\mathcal{W}$, a window Transformer block T_{win} models relationships among its cell features:

$$\{\mathbf{f}'_t\}_{t\in\mathcal{W}} = T_{\mathrm{win}}(\{\mathbf{f}_t\}_{t\in\mathcal{W}})\,. \tag{4}$$

We then shift the window partition and apply a second block to connect cells across the previous window boundaries (Liu et al., 2021). The updated cell features are reassembled into a grid, projected to the original channels, and added to G to produce $\widetilde{G}$ for subsequent along-segment readout.

3.4 PANOPTIC PREDICTION

To adapt image readout to the adopted line representation, we sample $\widetilde{G}$ at both endpoints and the midpoint of each constituent segment. Let $\mathbf{e}_j$ denote the image feature of segment j, obtained by projecting and averaging the features sampled at these three positions, and let $\mathcal{S}_i$ index the constituent segments of primitive p_i. We obtain its image feature $\mathbf{v}_i$ by summing the channel-wise mean and maximum over these segments:

$$\mathbf{v}_i = \mathrm{Mean}_{j\in\mathcal{S}_i}(\mathbf{e}_j) + \mathrm{Max}_{j\in\mathcal{S}_i}(\mathbf{e}_j). \tag{5}$$

We fuse the aligned point and image features of each primitive using a scalar gate conditioned on both representations (Luo et al., 2025).

Decoder. Following the feature-based query initialization in OneFormer3D, we initialize decoder queries from selected primitive features (Kolodiazhnyi et al., 2024). The refined queries predict primitive semantic labels and instance categories and masks. At inference, every primitive initializes a query and receives its corresponding semantic prediction.

Loss Function. During training, we jointly supervise the semantic and instance predictions with the objective $\mathcal{L} = \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{bce}}\mathcal{L}_{\text{bce}} + \lambda_{\text{dice}}\mathcal{L}_{\text{dice}}$. The semantic loss $\mathcal{L}_{\text{sem}}$ is cross-entropy on the final semantic predictions, and $\mathcal{L}_{\text{cls}}$ is cross-entropy on matched mask classes. The binary cross-entropy term $\mathcal{L}_{\text{bce}}$ and Dice term $\mathcal{L}_{\text{dice}}$ (Milletari et al., 2016) supervise mask membership and overlap, respectively. Each of the three instance loss terms sums contributions from all supervised decoder outputs, and each λ weights its corresponding loss.

At inference, we apply VecFormer's Branch Fusion Refinement (BFR) (Wei et al., 2025) to reconcile semantic and instance predictions and form the panoptic output.

4 Experiments

In this section, we first describe the experimental setup and compare our framework with existing methods on FloorPlanCAD and ArchCAD-Public. We then examine the main components, HPIF interaction directions, and POCE through ablations. Finally, qualitative comparisons illustrate the primitive-level semantic predictions for CAD drawings from both datasets.

4.1 Experimental Setup

Dataset. FloorPlanCAD is a benchmark for panoptic symbol spotting in real architectural floor plans, including residential buildings, schools, hospitals, and shopping malls. Its vector drawings preserve individual graphical primitives and provide semantic labels together with instance annotations for countable symbols. The drawings are cropped into regions of $10\,\text{m} \times 10\,\text{m}$. We follow the official split, with 6,965, 3,827, and 810 floor plan drawings in the training, test, and validation sets, respectively. The annotations cover 35 categories, comprising 30 thing classes and five stuff classes (Fan et al., 2021; Wei et al., 2025).

ArchCAD-Public is the publicly released subset of ArchCAD, with broader coverage of architectural elements, structural components, and drawing notations, as well as a larger collection of drawing regions than FloorPlanCAD. Each sample spans a larger drawing region of $14\,\text{m} \times 14\,\text{m}$. The public annotation scheme covers 30 foreground categories, including 21 thing classes and nine stuff classes. The training and test partitions contain 36,987 and 4,110 samples, respectively. Because the release does not provide an independent validation set, we randomly select 10% of the training samples for validation (Luo et al., 2025).

Metrics. Following FloorPlanCAD, we use Panoptic Quality (PQ) defined on vector graphics as the primary metric. It combines Recognition Quality (RQ) with Segmentation Quality (SQ) to evaluate both symbol recognition and primitive grouping. PQ is defined as (Fan et al., 2021)

$$\text{PQ} = \underbrace{\frac{|\text{TP}|}{|\text{TP}| + \frac{1}{2}|\text{FP}| + \frac{1}{2}|\text{FN}|}}_{\text{RQ}} \times \underbrace{\frac{\sum_{(s_p, s_g)\in \text{TP}} \text{IoU}(s_p, s_g)}{|\text{TP}|}}_{\text{SQ}}, \tag{6}$$

where s_p and s_g are predicted and ground-truth symbols represented as sets of primitives. TP contains matched symbol pairs, while FP and FN contain unmatched predictions and ground-truth symbols, respectively. A pair is matched when its semantic labels agree and its IoU exceeds 0.5. RQ measures recognition quality, and SQ is the mean IoU of matched pairs. We report overall PQ and its thing and stuff counterparts, denoted by PQ_{th} and PQ_{st}, respectively. Logarithmic weighting reduces the influence of very long primitives. The overlap is computed from primitive lengths, where $L(e)$ denotes the arc length of primitive e:

$$\text{IoU}(s_p, s_g) = \frac{\sum_{e\in s_p \cap s_g} \log(1 + L(e))}{\sum_{e\in s_p \cup s_g} \log(1 + L(e))}. \tag{7}$$

Implementation Details. We use PTv3 and UHRNet as the point and image backbones, respectively. The loss weight ratio is $\lambda_{\text{sem}} : \lambda_{\text{cls}} : \lambda_{\text{bce}} : \lambda_{\text{dice}} = 2 : 1 : 2 : 2$. On both datasets, we train with AdamW (Loshchilov & Hutter, 2019), weight decay 0.1, and an initial learning rate of 0.0001, using a global batch size of 8 on 8 NVIDIA RTX 5880 Ada GPUs. After training each branch independently for 50 epochs, we jointly optimize both branches and HPIF for 30 epochs, using a backbone learning rate of 0.00002. We then freeze these components and train PCT and ICT for 10 epochs.

4.2 Main Results

4.2.1 Quantitative Results on FloorPlanCAD

Table 1: Panoptic symbol spotting on FloorPlanCAD with and without priors.

Method	w/o Prior			w/ Prior		
	PQ ↑	PQ_{th} ↑	PQ_{st} ↑	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
PanCADNet (Fan et al., 2021)	59.5	65.6	58.7	—	—	—
CADTransformer (Fan et al., 2022)	68.9	78.5	58.6	—	—	—
GAT-CADNet (Zheng et al., 2022)	73.7	—	—	—	—	—
SymPoint (Liu et al., 2024a)	83.3	84.1	48.2	—	—	—
SymPoint V2 (Liu et al., 2024b)	83.2	85.8	49.3	90.1	90.8	80.8
CADSpotting (Yang et al., 2026)	—	—	—	88.9	89.7	80.6
DPSS (Luo et al., 2025)	86.2	88.0	64.7	89.5	90.4	79.7
VecFormer (Wei et al., 2025)	88.4	90.9	85.9	91.1	91.8	90.4
Ours	**90.3**	**92.5**	**87.8**	**92.7**	**94.2**	**90.7**

We compare our framework with existing methods for CAD panoptic symbol spotting on FloorPlanCAD (Table 1). Baseline PQ results are taken from Wei et al. (2025). We evaluate methods under two settings: with and without priors (such as layer identifiers and primitive colors). Although these priors provide useful recognition cues, we still emphasize evaluation without them because it more closely reflects practical situations where such metadata is unavailable or unreliable. Our method achieves the highest overall, thing, and stuff PQ among the reported results in both settings, reaching 92.7 and 90.3 overall PQ with and without priors, respectively.

Our framework complements primitive geometry with visual information through HPIF and POCE. Compared with VecFormer, it improves overall PQ by 1.6 points with priors, while thing and stuff PQ increase from 91.8 to 94.2 and from 90.4 to 90.7, respectively. Without priors, the gains reach 1.9 points in overall PQ, 1.6 in thing PQ, and 1.9 in stuff PQ, demonstrating the effectiveness of our framework for CAD symbol spotting without auxiliary metadata (Wei et al., 2025).

Table 2: Primitive semantic results on FloorPlanCAD.

Method	w/o Prior		w/ Prior	
	F1 ↑	wF1 ↑	F1 ↑	wF1 ↑
DeepLabv3+R101 (Chen et al., 2018)	68.8	71.4	—	—
CADTransformer (Fan et al., 2022)	82.2	80.1	—	—
SymPoint (Liu et al., 2024a)	86.8	85.5	—	—
SymPoint V2 (Liu et al., 2024b)	87.0	86.3	89.5	88.3
CADSpotting (Yang et al., 2026)	—	—	93.5	93.9
DPSS (Luo et al., 2025)	92.0	93.2	93.1	93.2
VecFormer (Wei et al., 2025)	—	—	93.8	92.2
Ours	**93.3**	**93.2**	**94.2**	**94.5**

Primitive-Level Semantic Quality. We evaluate the semantic prediction of each graphical primitive using F1 and length-weighted F1 (Table 2). Without priors, our method achieves 93.3 F1 and 93.2 wF1. With priors, it achieves 94.2 F1 and 94.5 wF1, exceeding VecFormer by 0.4 and 2.3 points, respectively. Qualitative comparisons in Section 4.4 illustrate the corresponding prediction differences (Wei et al., 2025).

4.2.2 Quantitative Results on ArchCAD-Public

Table 3: Panoptic and primitive semantic results on ArchCAD-Public.

Method	PQ ↑	PQ_{th} ↑	PQ_{st} ↑	F1 ↑	wF1 ↑
CADTransformer (Fan et al., 2022)	64.3	66.5	56.6	—	—
SymPoint (Liu et al., 2024a)	67.7	72.3	43.0	—	—
SymPoint V2 (Liu et al., 2024b)	76.6	80.0	61.0	—	—
DPSS (Luo et al., 2025)	82.9	86.3	78.4	—	—
VecFormer (Wei et al., 2025)	86.8	88.9	81.6	83.2	82.0
Ours	**90.6**	**94.0**	**83.1**	**89.3**	**88.4**

We compare our method with existing approaches for panoptic symbol spotting on ArchCAD-Public (Table 3). We train VecFormer to convergence on this subset and report its results alongside the other baselines from Luo et al. (2025). Following the evaluation setting of existing methods, we use no additional layer or color priors. This setting reflects practical applications where such metadata is unavailable and challenges the model to distinguish a broader range of architectural elements using geometry and visual appearance. Our method achieves the highest overall, thing, and stuff PQ among the reported methods, reaching 90.6, 94.0, and 83.1, respectively.

Compared with VecFormer, our framework improves overall PQ by 3.8 points on ArchCAD-Public, with gains of 5.1 in thing PQ and 1.5 in stuff PQ. The larger gain in thing PQ highlights stronger panoptic performance on countable symbol instances. These results support the effectiveness of combining primitive geometry with visual information across both CAD benchmarks, including recognition without auxiliary layer or color priors (Wei et al., 2025).

4.3 Ablation Study

Table 4: Ablation studies on FloorPlanCAD with layer priors.

(a) Main components

Point branch	Image branch	HPIF	POCE	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
✓	—	—	—	91.16	91.80	90.41
✓	✓	—	—	91.28	92.20	90.16
✓	✓	✓	—	92.37	93.88	90.41
✓	**✓**	**✓**	**✓**	**92.74**	**94.18**	**90.70**

(b) HPIF interaction directions

I2P	P2I	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
—	—	91.31	92.40	90.08
✓	—	92.57	94.01	90.57
—	✓	92.39	93.82	90.32
✓	**✓**	**92.74**	**94.18**	**90.70**

(c) POCE

PCT	ICT	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
—	—	92.37	93.88	90.41
—	✓	92.63	**94.19**	90.51
✓	—	92.66	94.10	90.70
✓	**✓**	**92.74**	94.18	**90.70**

Effects of Individual Components. We conduct a series of ablation experiments to evaluate the effects of individual components, as shown in Table 4(a). All variants in Table 4 follow the training

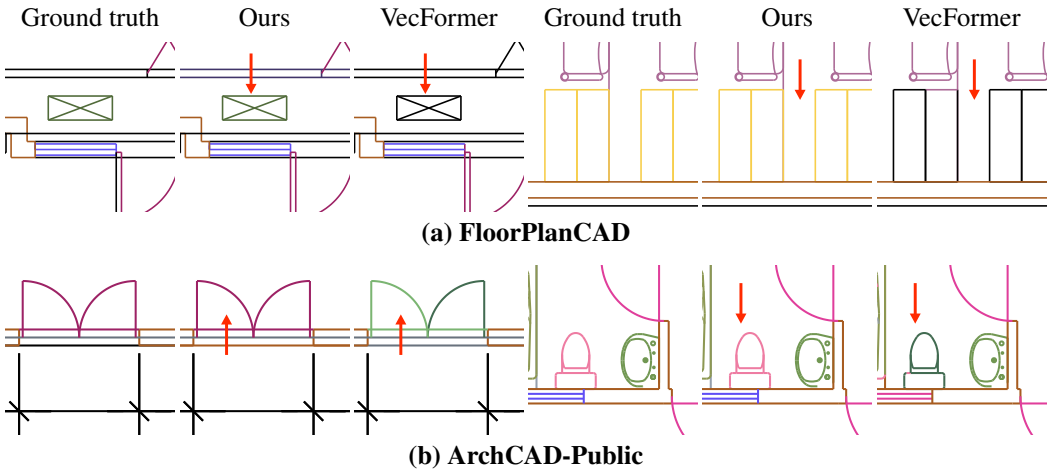


Figure 2: **Primitive-level semantic predictions** on FloorPlanCAD (top) and ArchCAD-Public (bottom). Each triplet shows ground truth, ours, and VecFormer.

protocol in Section 4.1 and are trained separately with the same pretrained weights for shared components, keeping all other settings fixed. Introducing HPIF raises PQ to 92.37, improving on the dual-branch model by 1.09 points and demonstrating the effectiveness of geometric–visual interaction during backbone encoding. POCE further improves the resulting representations through local context enhancement.

HPIF Interaction Directions. We examine the two directions of HPIF while retaining output fusion and PCT and ICT (Table 4(b)). Both directions improve PQ, with I2P yielding the stronger individual result. Bidirectional exchange achieves the highest PQ among these variants, supporting the complementary roles of the two exchange directions.

POCE. We evaluate PCT and ICT separately and together to examine the value of local context after backbone interaction (Table 4(c)). Both modules improve PQ individually, showing that local context remains useful after cross-modal feature exchange. Their combination achieves the highest overall PQ of 92.74, improving on the model without POCE by 0.37 points. Their combined result indicates that the two paths offer complementary benefits to the overall panoptic prediction.

Additional ablation studies are provided in Appendix A.

4.4 Qualitative Results

We compare primitive-level semantic predictions with VecFormer (Wei et al., 2025) on FloorPlanCAD and ArchCAD-Public in Figure 2. In these examples, our method produces more accurate semantic predictions than VecFormer. Semantic color maps and full views are provided in Appendix B.

5 Conclusion

We presented a point–image fusion framework for panoptic symbol spotting in CAD drawings. Hierarchical Point–Image Fusion integrates spatially aligned, bidirectional exchange into backbone feature extraction, enriching point representations with visual features and image representations with primitive geometry. POCE further aggregates nearby geometric and visual evidence to strengthen primitive representations. The framework achieves 92.7 PQ on FloorPlanCAD and 90.6 PQ on ArchCAD-Public, surpassing the compared state-of-the-art methods. These results establish the value of combining interaction during feature extraction with primitive context modeling for CAD panoptic recognition. As the benchmarks evaluated here operate on cropped CAD regions, future work will explore scalable inference on complete CAD drawings.

AI USE STATEMENT

Generative AI tools were used to assist with translation, language editing and polishing to improve the readability of the manuscript, literature retrieval and discovery, manuscript formatting, and the generation and revision of experimental code. All AI-assisted text and code were reviewed, revised, and verified by the authors. Literature and reference information suggested by AI tools was independently checked by the authors, and AI-assisted experimental code was tested and validated by the authors. The authors take full responsibility for the final manuscript, references, code, and experimental results.

REPRODUCIBILITY STATEMENT

The model architecture and feature interactions are described in Section 3. Section 4.1 specifies the datasets, evaluation metrics, optimization settings, and staged training schedule. Section 4.3 and Appendix A report ablation studies of the main components, interaction directions, image sampling strategies, number of interaction modules, and neighborhood radius. Additional qualitative comparisons and semantic color maps are provided in Appendix B.

REFERENCES

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, pp. 833–851, 2018.

Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. In *International Conference on Learning Representations*, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Zhiwen Fan, Lingjie Zhu, Honghua Li, Xiaohao Chen, Siyu Zhu, and Ping Tan. FloorPlanCAD: A large-scale CAD drawing dataset for panoptic symbol spotting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10128–10137, 2021.

Zhiwen Fan, Tianlong Chen, Peihao Wang, and Zhangyang Wang. CADTransformer: Panoptic symbol spotting transformer for CAD drawings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10986–10996, 2022.

Yan Gong, Bohao Li, Bowen Du, and Junchen Ye. Text-aided multi-modal panoptic symbol spotting for CAD floor plan drawings. *arXiv preprint arXiv:2607.12678*, 2026.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Alexander Kirillov, Kaiming He, Ross B. Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9404–9413, 2019.

Maxim Kolodiazhnyi, Anna Vorontsova, Anton Konushin, and Danila Rukhovich. OneFormer3D: One transformer for unified point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20943–20953, 2024.

Wenlong Liu, Tianyu Yang, Yuhan Wang, Qizhi Yu, and Lei Zhang. Symbol as points: Panoptic symbol spotting via point-based representation. In *International Conference on Learning Representations*, 2024a.

Wenlong Liu, Tianyu Yang, Qizhi Yu, and Lei Zhang. SymPoint revolutionized: Boosting panoptic symbol spotting with layer feature enhancement. *arXiv preprint arXiv:2407.01928*, 2024b.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Ruifeng Luo, Zhengjie Liu, Tianxiao Cheng, Jie Wang, Tongjie Wang, Fei Cheng, Fu Chai, Yanpeng Li, Xingguang Wei, Haomin Wang, Shenglong Ye, Wenhai Wang, Zhang Zhang, Yu Qiao, Hongjie Zhang, and Xianzhong Zhao. ArchCAD-400K: A large-scale CAD drawings dataset and new baseline for panoptic symbol spotting. In *Advances in Neural Information Processing Systems*, pp. 127715–127739, 2025.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *Proceedings of the 4th International Conference on 3D Vision*, pp. 565–571, 2016.

Zhiliang Peng, Wei Huang, Shanzhi Gu, Lingxi Xie, Yaowei Wang, Jianbin Jiao, and Qixiang Ye. Conformer: Local features coupling global representations for visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 367–376, 2021.

AJ Piergiovanni, Vincent Casser, Michael S. Ryoo, and Anelia Angelova. 4D-Net for learned multi-modal alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15435–15445, 2021.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.

Alireza Rezvanifar, Melissa Cote, and Alexandra Branzan Albu. Symbol spotting for architectural drawings: state-of-the-art and new industry-driven developments. *IPSJ Transactions on Computer Vision and Applications*, 11:2, 2019.

Alireza Rezvanifar, Melissa Cote, and Alexandra Branzan Albu. Symbol spotting on digital architectural floor plans using a deep learning-based framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2419–2428, 2020.

Jian Wang, Xiang Long, Guowei Chen, Zewu Wu, Zeyu Chen, and Errui Ding. U-HRNet: Delving into improving semantic representation of high resolution network for dense prediction. *arXiv preprint arXiv:2210.07140*, 2022.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, 2021.

Xingguang Wei, Haomin Wang, Shenglong Ye, Ruifeng Luo, Zhang Zhang, Lixin Gu, Jifeng Dai, Yu Qiao, Wenhai Wang, and Hongjie Zhang. Point or line? Using line-based representation for panoptic symbol spotting in CAD drawings. In *Advances in Neural Information Processing Systems*, pp. 50036–50061, 2025.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point Transformer V3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4840–4851, 2024.

Chunlong Xia, Xinliang Wang, Feng Lv, Xin Hao, and Yifeng Shi. ViT-CoMer: Vision transformer with convolutional multi-scale feature interaction for dense predictions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5493–5502, 2024.

Bingchen Yang, Haiyong Jiang, Hao Pan, and Jun Xiao. VectorFloorSeg: Two-stream graph attention network for vectorized roughcast floorplan segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1358–1367, 2023.

Fuyi Yang, Jiazuo Mu, Mingqian Zhang, Yanshun Zhang, Junxiong Zhang, Lan Xu, and Yingliang Zhang. CADSpotting: Robust panoptic symbol spotting on large-scale CAD drawings. *Computational Visual Media*, 12(4):1127–1149, 2026.

Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. DeepInteraction: 3D object detection via modality interaction. In *Advances in Neural Information Processing Systems*, pp. 1992–2005, 2022.

Zhaohua Zheng, Jianfang Li, Lingjie Zhu, Honghua Li, Frank Petzold, and Ping Tan. GAT-CADNet: Graph attention network for panoptic symbol spotting in CAD drawings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11747–11756, 2022.

A Additional Ablation Studies

A.1 Image Sampling

Table 5: Image sampling strategies.

Sampling	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
Center	92.47	94.14	90.19
3 points/segment	**92.74**	**94.18**	**90.70**
5 points/segment	92.65	94.14	90.66

We compare sampling at primitive centers with sampling at three or five positions along each segment (Table 5). Three positions achieve the highest PQ of 92.74, with a 0.51-point gain in stuff PQ over center sampling, supporting the value of visual evidence along constituent segments. This design also achieves higher PQ than five-position sampling with fewer samples.

A.2 Number of HPIF Modules

Table 6: Results with different numbers of HPIF modules.

HPIF modules	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
2	92.19	93.89	90.03
4	92.48	93.99	90.32
8	**92.74**	**94.18**	**90.70**

We compare two, four, and eight HPIF modules placed symmetrically across the backbones, keeping all other settings fixed (Table 6). PQ increases with the number of interaction modules, reaching 92.74 with eight modules, a gain of 0.55 points over two modules. This result supports exchanging geometric and visual information throughout feature extraction.

A.3 Neighborhood Radius

Table 7: PCT neighborhood radius ($K = 32$).

Radius r	PQ ↑	PQ_{th} ↑	PQ_{st} ↑
0.01	92.60	94.05	90.51
0.03	**92.74**	**94.18**	**90.70**
0.10	92.68	94.08	90.65

We examine the sensitivity of the full model to the PCT neighborhood radius r, fixing $K = 32$ and training each radius setting separately (Table 7). Across $r \in \{0.01, 0.03, 0.10\}$, PQ ranges from 92.60 to 92.74, showing limited sensitivity over the tested range. Our default setting, $r = 0.03$, achieves 92.74 PQ.

B Additional Qualitative Results

B.1 Semantic Color Maps

Figures 3 and 4 show the category-to-color mappings used in our qualitative comparisons. The FloorPlanCAD display groups follow SymPoint and VecFormer (Liu et al., 2024a; Wei et al., 2025), while the ArchCAD-Public categories are arranged into five groups for visualization. Within each dataset, the same color mapping is used for ground truth and predictions from both methods.

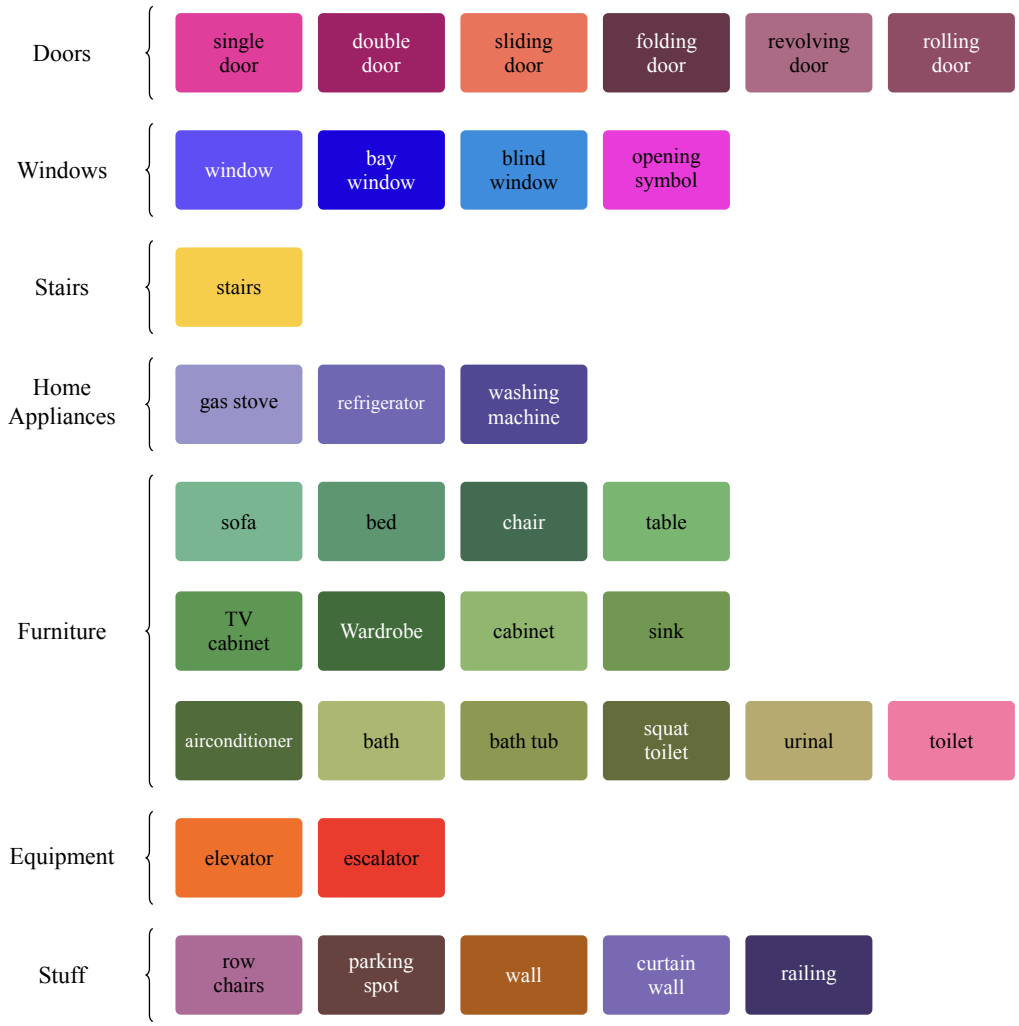


Figure 3: Semantic color map for FloorPlanCAD, covering 35 categories.



Figure 4: Semantic color map for ArchCAD-Public, covering 30 categories.

B.2 Full-View Comparisons

We provide full views of the samples shown in Figure 2 to complement the close-up comparisons.

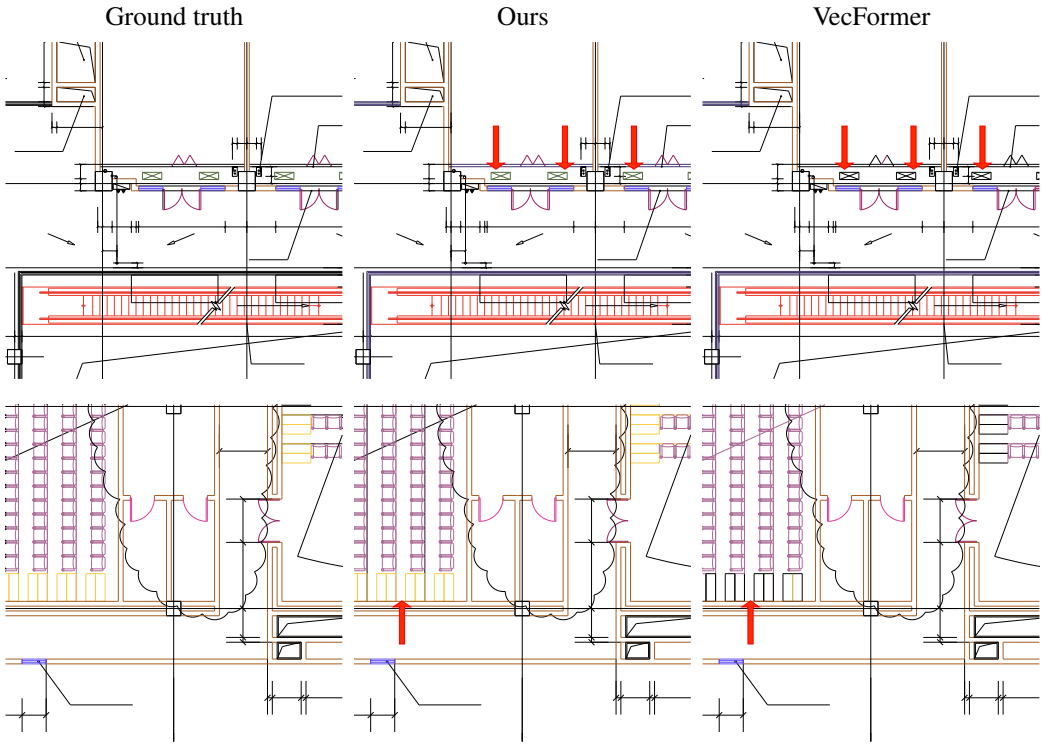


Figure 5: Full-view primitive-level semantic predictions on FloorPlanCAD. Columns show ground truth, ours, and VecFormer.

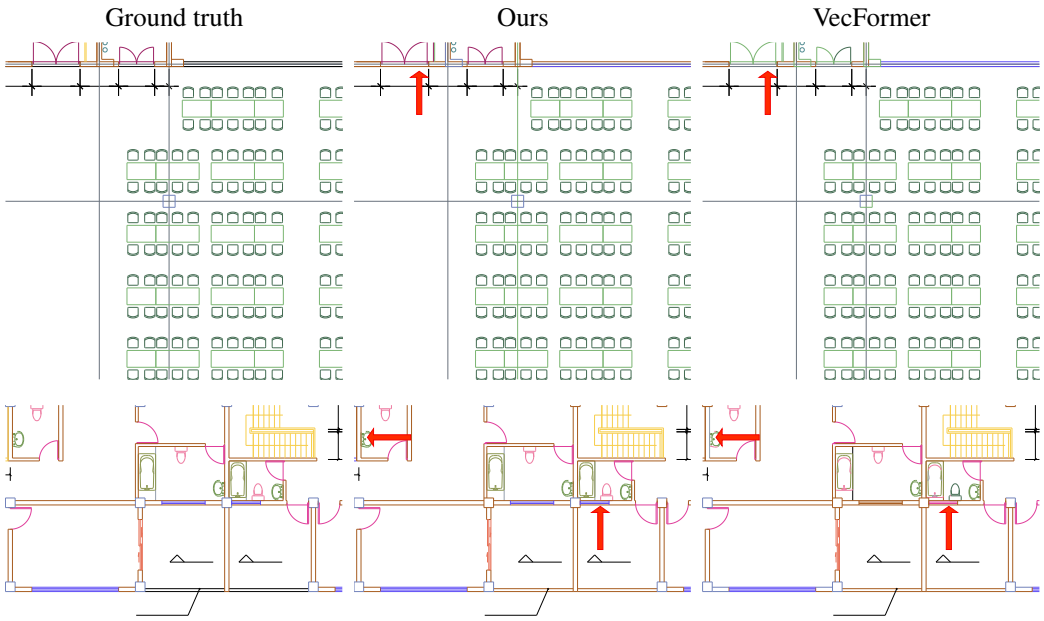


Figure 6: Full-view primitive-level semantic predictions on ArchCAD-Public. Columns show ground truth, ours, and VecFormer.