# MV-GEL: Language-Driven Multi-View Geometric Entity Localization on Meshes

Kartik Bali[1,2] and Roland Aydin[3]

[1] Helmholtz Zentrum Hereon, Max-Planck-Strasse 1, 21502 Geesthacht, Germany
[2] Institute for Continuum and Material Mechanics, Hamburg University of Technology, Eissendorfer Strasse 42, 21073 Hamburg, Germany
[3] German Research Center for Artificial Intelligence, Campus D3 2, 66123 Saarbrücken, Germany
kartik.bali@hereon.de, roland.aydin@dfki.de

**Abstract.** Identifying and grounding precise geometric entities, such as edges, planar regions, and curved surfaces within 3D objects, is foundational to computer-aided design (CAD), robotic manipulation, and scientific simulation. Although modern Vision Language Models (VLMs) have advanced referring segmentation (RIS) in the image domain, extending such language-driven localization to structured 3D geometry is substantially harder. The 3D object appearance is highly sensitive to viewpoints; a single perspective may render a target entity clearly observable, while another may suffer from severe occlusion or foreshortening. In this work, we attempt to solve these challenges with **MV-GEL** (Multi-View Geometric Entity Localization), a framework for localizing fine-grained geometric entities on polygon meshes from natural language queries. Our key insight is that reliable CAD entity (i.e., faces, edges or solids) localization depends on selecting views that make the queried entity maximally interpretable. We introduce *GELviews*, a prompt-conditioned ranking module that prioritizes viewpoints based on language prompted observability of geometric CAD entities. Selected views are processed by a VLM-based reasoning segmentation backbone, and predicted masks are lifted to the corresponding meshes via geometry-aware ray casting. Our framework is completely CAD agnostic and relies only on 3D meshes. Experiments show up to a 1.7× improvement in face-level IoU and over 4.5× gains in edge-level F1 compared to vanilla baselines, substantially outperforming CLIP-based and random view sampling, particularly for thin and view-sensitive structures. The dataset, code and trained checkpoints are available at https://github.com/kbali1297/MV-GEL.

**Keywords:** Multimodal LLMs · 3D-understanding · CAD

## 1 Introduction

Three-dimensional semantic localization is a fundamental prerequisite for intelligent systems to perceive, reason, and act in real-world environments. While classical approaches emphasize global pose estimation or reconstruction, modern applications demand precise identification of semantically meaningful object

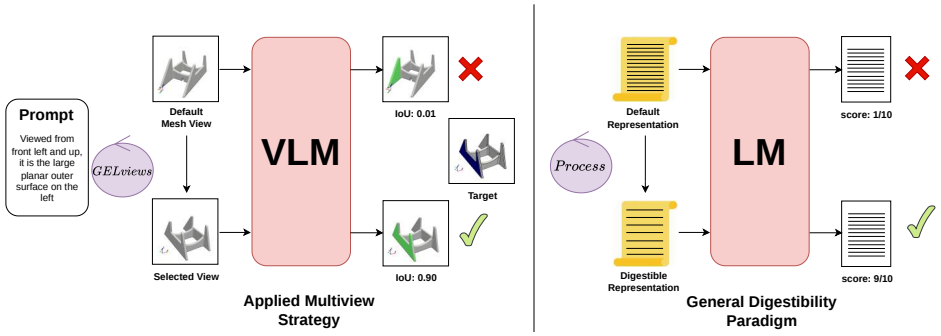



**Fig. 1:** We can tie our multi-view strategy to a more general digestibility paradigm, where certain representations of data are more interpretable to a language model than others

parts. In robotics, grasp planning depends on functional regions such as handles; in CAD and manufacturing, simulation and editing require selecting exact geometric entities including fillets, chamfers, and load-bearing surfaces. These targets are defined as discrete topological elements present in a 3D structure and their spatial identification is crucial for the relevant task.

Recent vision-language models have demonstrated strong semantic reasoning in images [2, 5, 31, 43], and a few have explored reasoning-based segmentation [25, 41, 51]. However, using these frameworks for 3D segmentation and entity localization depends critically on how visual information is presented to the VLM. In 3D settings, the same underlying geometry can appear either highly informative or severely ambiguous depending on the viewpoint. Self-occlusion, depth conflation, and thin structural elements often render relevant entities visually indigestible from arbitrary projections. Thus, language-driven geometric localization is not merely a segmentation problem, but a question of making structured 3D information interpretable to current VLMs that process visual information in 2D.

In this work, we address this visual interpretability problem for VLMs in the context of entity localization for mesh objects, derived from CAD. We introduce a prompt-conditioned ranking framework that estimates entity-specific geometric visibility across candidate views, presenting the reasoning segmentation model with geometrically digestible evidence that helps it achieve accurate entity localization, as illustrated in Fig. 1. In summary, our contributions are threefold:

- We create a dataset and formalize the task of locating geometric entities in CAD meshes guided via natural language.
- We propose *GELviews*, a prompt-conditioned view ranking framework that prioritizes viewpoints given to reasoning segmentation 2D VLMs, based on entity-specific geometric visibility.

– We introduce a CAD-agnostic framework **MV-GEL** for localizing these geometric entities using text prompts directly on the mesh.

We evaluate the results of our framework by introducing a mesh-aware lifting strategy that maps 2D segmentation masks to discrete topological mesh entities (mesh faces and edges).

## 2 Related Work

### 2.1 Vision-Language Models and Referring Segmentation

The integration of visual and textual modalities has driven remarkable progress in open-vocabulary image understanding. Foundational vision-language models (VLMs) like CLIP [40], ALIGN [18], and BLIP-2 [28] align image and text embeddings, enabling zero-shot recognition. Building on this, Large Multimodal Models (LMMs) such as LLaVA [31], InstructBLIP [7], and Qwen-VL [2] have demonstrated sophisticated visual reasoning capabilities.

In parallel, referring expression segmentation (RES) has evolved from relying on complex multi-modal fusion networks [8, 30, 46, 50] to leveraging foundation models like the Segment Anything Model (SAM) [23, 53, 56]. Successors like LISA [25], PixelLM [41], Osprey [51], and GSVA [48] further improve semantic segmentation via pixel-level dense reasoning using Vision Language Models. However, these models are trained predominantly on natural images (e.g., COCO, LVIS) and struggle to comprehend the rigid geometry, thin structures, and geometric description cues inherent to Computer-Aided Design (CAD). Our work bridges this domain gap by explicitly fine-tuning a reasoning VLM on CAD mesh renderings and geometry in natural language used to spatially describe these features.

### 2.2 Language Guided Segmentation in 3D

Analyzing 3D geometry via 2D projections is a highly effective paradigm, popularized by MVCNN [42] and expanded by recent view-graph and transformer architectures [10, 45]. Recently, the field has shifted toward lifting 2D VLM features into 3D space to achieve open-vocabulary 3D understanding. Techniques such as ConceptFusion [15], OpenScene [38], and LERF [21] project CLIP features onto point clouds or Neural Radiance Fields (NeRFs). For part-level segmentation, PartSLIP [32], PartSLIP++ [54], and SATR [1] leverage 2D grounding cues and GLIP [29] to localize semantic parts in 3D point clouds. PointCLIP [52] and PointCLIPV2 [55] adapt vision-language models to point clouds through multi-view rendering and feature alignment, while COPS [9] and GeoZE [37] further improve open-vocabulary understanding via geometry-aware aggregation of 2D visual features and geometric priors. More recent approaches, such as PARIS3D [20], PointBind [11], ReferSplat [14], and IPDN [3], incorporate multimodal reasoning, structural prompts, and multi-view constraints for language-guided 3D localization. Recent foundation-model-based methods such

as Find3D [36] and PatchAlign3D [13] learn language-aligned point- and patch-level representations for open-vocabulary part retrieval and segmentation. While these methods achieve impressive results on unstructured point clouds, Gaussian splats, and scene-level 3D representations, adapting them to fine-grained mesh entities with explicit topological connectivity remains challenging.

### 2.3 Deep Learning on CAD and Boundary Representations

Unlike standard meshes or point clouds, CAD models are defined by Boundary Representations (B-Reps), i.e graphs of parametric curves and surfaces. Early geometric deep learning adapted to B-Reps through UV-Net [17], BRepNet [26], and SB-GCN [19] for tasks like solid classification and machining feature recognition, heavily utilizing the ABC Dataset [24] and Fusion 360 Gallery [47].

Recently, text-driven 3D generation has extended into the CAD domain. Works like Text2CAD [22], CAD-LLaMA [27], and SolidGen [16] generate CAD construction sequences from text. Furthermore, B-repLer [33] and CAD-MLLM [49] introduced semantic editing of B-Reps using language models. Despite these advances, the fine-grained localization of arbitrary, language-described CAD entities (e.g., "the inner chamfered edge") remains largely unexplored. We address this by providing a unified pipeline for language-driven geometric entity localization on meshes.

### 2.4 Prompt-Aware View Selection

In multi-view pipelines, selecting optimal viewpoints is critical for information gain. While traditional Next-Best-View (NBV) algorithms rely on geometric heuristics [6] or deep reinforcement learning [4], the advent of VLMs has shifted the focus toward semantic view selection. Methods like Cap3D [35] and ViewRefer [12] dynamically rank camera poses by maximizing global CLIP similarity between 2D renderings and a language prompt. However, global semantic matching fails on homogeneous, textureless industrial CAD parts, where pinpointing fine-grained features (e.g., "the inner chamfered edge") requires explicit geometric awareness. Furthermore, while recent 3D referring segmentation methods tackle localized spatial understanding, they predominantly operate on unstructured point clouds or voxels, and are not designed to output exact topological entities defined by a mesh graph.

Our work bridges these disconnected domains. To the best of our knowledge, MV-GEL is the first framework to address prompt-conditioned, multi-view localization of exact topological entities on 3D meshes.

## 3 Problem Formulation

Let $\mathcal{M} = (V, \mathcal{E}, \mathcal{F})$ denote a discrete 3D triangle mesh, where $V = \{\mathbf{v}_i \in \mathbb{R}^3\}_{i=1}^{N}$ is the set of vertices, $\mathcal{E} = \{e_k\}_{k=1}^{K}$ is the set of edges, and $\mathcal{F} = \{f_j\}_{j=1}^{M}$ is the set of triangular faces. In the context of computer-aided design, a *geometric entity*

refers to a distinct topological element, for instance, a continuous surface patch or a boundary curve.

Given a natural language query $q$ describing a specific geometric entity (e.g., "the inner cylindrical surface", "the outer chamfered edge"), our goal is to localize the exact subset of mesh elements corresponding to this queried feature. Crucially, while the query conceptually describes an analytical CAD entity (a native B-Rep face or edge), the system is provided only the unstructured polygon mesh $\mathcal{M}$ as its geometric input at inference.

Formally, let $\mathcal{X} \in \{\mathcal{E}, \mathcal{F}\}$ represent the target domain of the queried entity type (edges or faces). We formulate geometric entity localization as predicting a binary labeling function $\hat{y} : \mathcal{X} \to \{0, 1\}$, where $\hat{y}(x_i) = 1$ indicates that the discrete mesh element $x_i \in \mathcal{X}$ belongs to the target structure described by $q$. The objective is to produce a topologically consistent prediction $\hat{y}$ over $\mathcal{M}$ such that the selected discrete mesh elements accurately reconstruct the continuous geometric entity intended by the user.

## 4 Dataset Construction

We construct a mesh–language localization dataset derived from the ABC CAD dataset. We sample entities and generate over 54k natural query-geometric entity pairs from over 5,000 CAD geometries spanning diverse mechanical parts with varying topology and surface complexity. For each CAD model, we randomly sample a single boundary representation (B-rep) primitive, restricted to either one surface (face) or one edge. As highlighted in the previous section, the sampled CAD entities provide the target views and segmentation maps during training, given the mesh $\mathcal{M}$ and the input query $q$.

**Multi-View Rendering and Ranking** For each CAD mesh $\mathcal{M}$, we generate a discrete set of candidate camera poses $\mathcal{V} = \{v_i\}$ distributed across uniform azimuth and elevation intervals. To evaluate a pose $v_i$ with respect to a target geometric entity $\mathcal{E}$ (an edge or a face), we first select a representative subset of points $\mathcal{P}_\mathcal{E} \subset \mathcal{E}$ using Farthest Point Sampling (FPS). We estimate the geometric observability of $\mathcal{E}$ from pose $v_i$ by casting rays from the camera center through the virtual image plane toward each point $p_j \in \mathcal{P}_\mathcal{E}$. A point is visible if the ray intersection is unobstructed by the mesh surface. We define the visibility score as $S(v_i, \mathcal{E}) = \sum_{p_j \in \mathcal{P}_\mathcal{E}} \mathbf{1}\{p_j \text{ is visible from } v_i\}$.

Because our visibility testing is computed from the camera's perspective, views where the entity is viewed at a slanted or grazing angle are naturally penalized. In such configurations, rays are more likely to encounter self-occlusion or grazing intersection errors, leading to a reduced count of visible points in $S(v_i, \mathcal{E})$. Consequently, our ranking mechanism inherently prioritizes "head-on" viewpoints that provide clear, unobstructed geometric exposure of the entity. Finally, for the top-ranked poses, we generate two synchronized renderings: an unannotated RGB image for model inference and a secondary mask-highlighted rendering for supervised training.

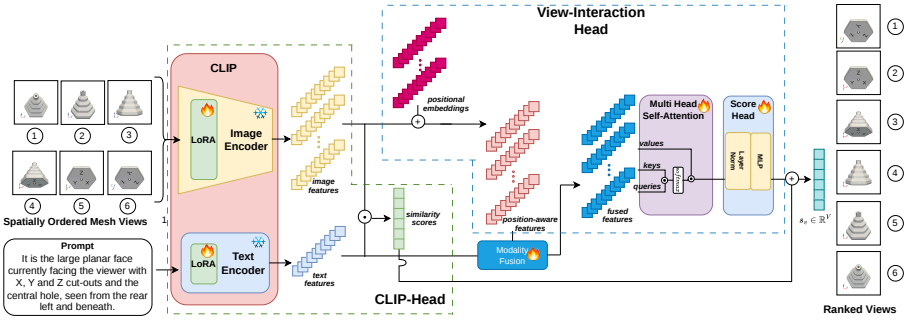



**Fig. 2:** *GELviews*: Our View selection framework takes the natural language prompt containing the geometric and view-point descriptions and returns semantically optimal views. In this example, the views with planar face at the bottom with cut-outs get higher ranks than the top views.

**Language Query Construction** For each sampled entity's maximal view, we construct a natural language query that uniquely identifies its geometric entity using only intrinsic geometric and topological cues, similar to how humans describe geometric entities. The descriptions rely primarily on properties such as curvature, orientation, adjacency, symmetry, continuity, and relative position within the part. This formulation enforces geometry-aware and viewpoint-specific grounding. Detailed data generation prompts can be found in the supplementary.

## 5 Method

Given a triangle mesh $\mathcal{M}$ and a natural language query $q$ describing a geometric structure, our objective is to localize the subset of mesh topological entities corresponding to the queried entity.

### 5.1 Prompt-Conditioned Geometric View Ranking (*GELviews*)

We formulate view selection as a prompt-conditioned geometric ranking problem. Given a set of candidate viewpoints, the objective is to prioritize views that maximize geometric observability of the queried entity. Unlike global semantic alignment approaches, this formulation requires reasoning about structural visibility, occlusion, and entity-specific spatial exposure across views.

**CLIP Head** Each rendered image $I_v$ is encoded using a CLIP vision encoder $\mathcal{E}_{img}$, and the query $q$ is encoded using the CLIP text encoder $\mathcal{E}_{txt}$: $f_v = \mathcal{E}_{img}(I_v), t = \mathcal{E}_{txt}(q)$, where $f_v, t \in \mathbb{R}^D$ are the global aligned feature embeddings (CLS token embeddings). These embeddings represent high-level semantic summaries of the view and the query, respectively, and are aligned due to the

CLIP training objective. We noticed that base CLIP image and text embeddings are not optimized to contain geometric semantic information and yield poor view selection when obtained from the base CLIP encoders. To mitigate this, both encoders are adapted using low-rank parameter updates, allowing task-specific specialization while preserving the pretrained vision–language alignment prior. This ensures stable optimization and prevents catastrophic forgetting of CLIP's semantic structure. As shown in the Fig. 2, for an initial estimate of view relevance, we compute normalized cosine similarity between each view embedding and the query embedding: $s_v^{\text{clip}} = \left\langle \frac{f_v}{\|f_v\|}, \frac{t}{\|t\|} \right\rangle$.

**View Positional Encoding** To incorporate geometric context, specifically to make the network aware of view-points and their geometric order, we inject learnable positional embeddings $p_v \in \mathbb{R}^D$ encoding the identity and relative ordering of each camera pose: $\tilde{f}_v = f_v + p_v$.

**Modality Fusion** While $s_v^{\text{clip}}$ measures independent alignment, view ranking requires conditioning image features explicitly on the query semantics. We therefore apply a fusion operator $\mathcal{F}$ between the view features and the global text embedding:$\hat{f}_v = \mathcal{F}(\tilde{f}_v, t)$. We investigate two complementary conditioning mechanisms: (i) FiLM [39], which performs feature-wise affine modulation $\text{FiLM}(\tilde{f}_v, t) = \gamma(t) \odot \tilde{f}_v + \beta(t)$, and (ii) Cross-Attention [34, 44], which enables token-level interaction between view features and the text embedding. FiLM provides lightweight global semantic conditioning and strictly generalizes additive and weighted-add fusion schemes as special cases (via learned scaling $\gamma(t)$ and bias $\beta(t)$), while preserving the pretrained CLIP representation at initialization. In contrast, Cross-Attention enables more expressive multimodal interaction by allowing query-dependent reweighting across feature dimensions.

**View Interaction Head** Individual view scoring ignores the fact that geometric visibility is inherently relative across viewpoints. For example, a surface may only be fully visible in one view while partially occluded in others. To capture such dependencies, we process the fused features using a multi-layer Transformer encoder $\mathcal{T}$ operating along the view dimension: $h_v = \mathcal{T}(\{\hat{f}_v\}_{v=1}^V)_v$. This self-attention mechanism allows each view to attend to all other views, enabling context-aware comparison and geometric reasoning. As a result, the representation of each view becomes informed by the global distribution of viewpoints. A lightweight scoring head maps the context-aware features to scalar geometric adjustment scores: $s_v^{\text{geo}} = \mathbf{w}^\top \text{LN}(h_v)$, where LN denotes LayerNorm. This component learns to correct or refine the initial semantic similarity based on geometric exposure and cross-view reasoning. The final ranking score combines semantic alignment and geometry-aware adjustment $s_v = \alpha \left(s_v^{\text{clip}} + s_v^{\text{geo}}\right)$, where $\alpha$ is a learnable temperature parameter controlling score sharpness. Views are ranked according to $s_v$, and the top-$K$ views are selected greedily from the *GELviews* scores.

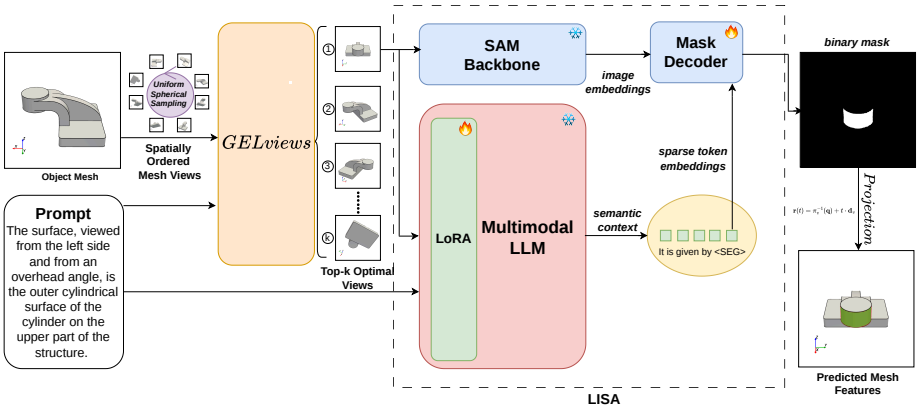



**Fig. 3:** *MV-GEL Pipeline*: In our framework, we use *GELviews* to compute the top-k views most relevant to localize a feature mentioned in the prompt. These views are then given sequentially to LISA-CAD for semantic segmentation to obtain view binary masks. Finally these binary masks are projected to the final mesh to obtain mesh features (edges/faces).

**Pairwise Ranking Objective** *GELviews* is trained to prioritize views that best expose the queried geometric entity. For each sample, we compute a visibility score $r_v$ per view and define $\mathcal{V}^+$ as the top-$k$ views, where $k = \max(1, \lfloor V \cdot \rho \rfloor)$ with a small fraction $\rho$. All remaining views form $\mathcal{V}^-$. This reduces ranking to separating geometrically strong views from weaker ones. Given predicted scores $s_v$, we enforce a margin $m$ between positive and negative views. For all $(i, j)$ with $i \in \mathcal{V}^+$ and $j \in \mathcal{V}^-$, we minimize

$$\mathcal{L}_{rank} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \log\left(1 + \exp\left(-(s_i - s_j - m)\right)\right), \tag{1}$$

where $\mathcal{P}$ is the set of positive–negative pairs. This soft margin ($m = 0.1$) objective encourages separation of informative viewpoints while avoiding noisy supervision over fine-grained ordering among weak views.

**View Non-Maximum Suppression** The ranked views produced by *GELviews* may contain angularly redundant viewpoints that observe nearly identical surface regions. In our experiments we realised that applying view-NMS, contrary to popular practices, hurts our metrics. This is because views nearby act as fallback for entity segmentation (especially in cases where VLM might miss them on the main view) and thus aid in recalling the entities as views monotonically increase. As a result we have omitted it from our main analysis. Readers are advised to have a look at Table 2, where the ViewNMS version for top-3 views (for angle threshold 45°) performs much worse than the regular greedy selection.

### 5.2 Language-Driven Segmentation with LISA

We adopt LISA [25] as the underlying language-conditioned segmentation backbone. The model operates in image space and predicts a binary mask conditioned on a special `<SEG>` token, whose hidden representation is projected into the SAM mask decoder as a sparse prompt embedding. While segmentation and training supervision is performed on 2D ground truth masks, evaluation is defined on CAD mesh entities via a geometry-aware lifting mechanism.

We retain the original LISA training objectives, including the autoregressive language modeling loss and mask supervision (binary cross-entropy and DICE).

### 5.3 Modular Multi-View Localization Pipeline

We now summarize the full MV-GEL pipeline, in Fig. 3, in functional form. Given a mesh $\mathcal{M}$ and query $q$, we first render a set of calibrated views $I_k = \mathcal{R}(\mathcal{M}, \pi_k), \quad k = 1, \ldots, V$, where $\pi_k$ denotes camera parameters and $\mathcal{R}$ is a rendering operator.

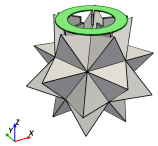
*View Selection.* The prompt-conditioned ranking module *GELviews* selects a subset of informative views $\mathcal{S} = \Psi(\{I_k\}_{k=1}^{V}, q)$, where $\mathcal{S} \subset \{1, \ldots, V\}$ contains the top-$K$ geometrically relevant viewpoints.

*Image-Space Segmentation.* For each selected view $k \in \mathcal{S}$, the segmentation backbone predicts a binary mask $m_k = \Gamma(I_k, q)$, where $\Gamma$ denotes the LISA-based segmentation model operating in image space.

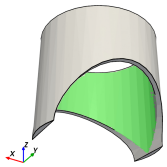
*Geometric Lifting.* The predicted masks are lifted to mesh topology via ray projection, producing a 3D labeling $\hat{y} = \mathcal{L}(\{m_k\}_{k\in\mathcal{S}}, \{\pi_k\}_{k\in\mathcal{S}}, \mathcal{M})$, where $\mathcal{L}$ denotes the geometry-aware lifting operator mapping foreground pixels to mesh faces or edges. The overall localization is therefore obtained by composition $\hat{y} = \mathcal{L} \circ \Gamma \circ \Psi(\mathcal{M}, q)$, with *GELviews* ($\Psi$) and the segmentation backbone ($\Gamma$) trained independently. This modular design preserves pretrained segmentation priors and avoids joint optimization through the rendering process.

## 6 Experiments

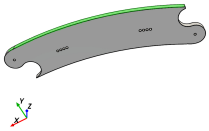
**Training** We train *GELviews* on 54k language queries paired with annotated top-performing views. LoRA adapters were used to fine tune clip image and text encoders with r=8 and alpha=16, with a layer dropout equal to 0.1. For each query, we uniformly sample candidate viewpoints over elevations in $[-60°, 60°]$ and azimuths in $[0°, 360°)$ at $30°$ increments, yielding 60 candidate views per instance. The ranking module is optimized to prioritize views that maximize entity-specific geometric observability. We place the top 5 ($\rho = 0.09$) views in the $\mathcal{V}^+$ set and the rest in $\mathcal{V}^-$. For segmentation, we fine-tune `LLaVA-7B` in the LISA-CAD setting using the same 54k query–binary mask pairs. Training is



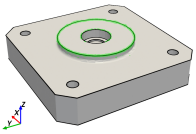
**(a)** Looking at it from the front-left corner and a higher vantage point, the surface is the top annular ring face with top axis parallel to z-axis surrounding the circular hole.



**(b)** It is the inner surface of the twisted band-continuous concave interior surface extending from the upper inner rim down to the lower pointed inner opening, most clearly seen from front-left looking up at it.



**(c)** The edge, with viewpoint shifted towards rear-left and viewed from below, is the outer long convex longitudinal edge of the plate.

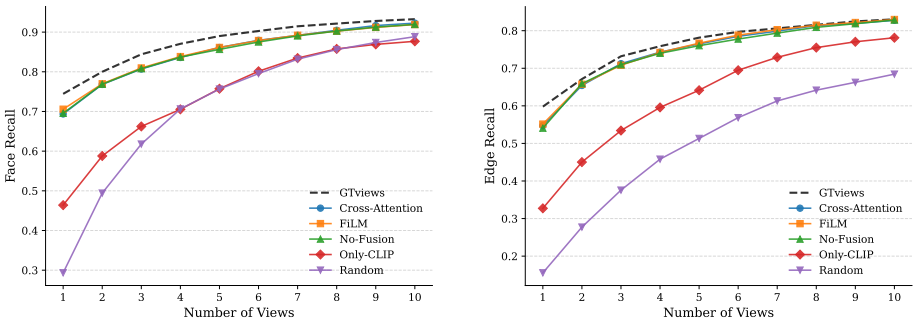


**(d)** From the left side of the rear face and from a superior vertical position, its the circular edge at the base of the raised boss where it meets the top surface of the plate.

**Fig. 4:** Examples of language-driven face and edge localization. The green mask denotes entity segmentation on the meshes via our **MV-GEL** framework.

conducted on 4 NVIDIA H100 GPUs, updating only the LoRA adapter layers and the mask decoder while keeping the remaining backbone parameters frozen. We use a learning rate of $1 \times 10^{-4}$ for all our trainable modules. Evaluation is performed on a held-out test set of 1535 query–entity pairs spanning 218 CAD meshes. After topology-aware lifting, performance is reported using face and edge-level metrics computed directly on the CAD mesh.

**Evaluation Strategy** We evaluate localization directly on mesh entities rather than in image space. Predictions and ground truth are represented as sets of faces or edges, and overlap is computed using geometric weighting to reduce variations due to mesh discretization. Let $\mathcal{S}_p$ and $\mathcal{S}_g$ denote predicted and ground-truth entity sets, and let $w(s)$ denote the geometric weight of an entity (triangle area for faces, edge length for edges). Weighted intersection and union are defined as $W_{\text{int}} = \sum_{s \in \mathcal{S}_p \cap \mathcal{S}_g} w(s), W_{\text{union}} = \sum_{s \in \mathcal{S}_p \cup \mathcal{S}_g} w(s)$. Further we define metrics such as Intersection over Union (IoU), Precision and Recall on the mesh as follows: $\text{IoU} = \frac{W_{\text{int}}}{W_{\text{union}}}, \text{Precision} = \frac{W_{\text{int}}}{\sum_{s \in \mathcal{S}_p} w(s)}, \text{Recall} = \frac{W_{\text{int}}}{\sum_{s \in \mathcal{S}_g} w(s)}$. This unified formulation applies to both faces and edges by simply modifying the geometric




**(a)** Face Recall **(b)** Edge Recall

**Fig. 5:** Impact of view quantity on recall performance for Faces (a) and Edges (b) using CAD adapted LISA.

weight definition. The use of area and length weighting ensures that evaluation reflects true geometric coverage and structural correctness.

**Table 1:** Performance evaluation of domain adapted and unadapted LISA variants and view selectors across Face@ top1 views and Edge@ top1 views metrics. *GTviews*, short for the Ground Truth views, act as the oracle target views of CAD meshes on the test set: the single annotated view each entity was marked on and to which it's referring caption corresponds. Test set of 1535 queries (655 faces, 880 edges) from 218 meshes.

| | | | **Face@ top1 views** | | | | **Edge@ top1 views** | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **S.no** | **Seg. VLM** | **View Selector** | **IoU** | **Prec.** | **Rec.** | **F1** | **IoU** | **Prec.** | **Rec.** | **F1** |
| 1 | LISA (Vanilla) | Cross-Attention | 0.292 | 0.337 | **0.652** | 0.390 | 0.043 | 0.048 | 0.494 | 0.074 |
| 2 | LISA (Vanilla) | FiLM | **0.293** | **0.342** | 0.647 | **0.391** | **0.046** | **0.053** | **0.503** | **0.079** |
| 3 | LISA (Vanilla) | No-Fusion | 0.293 | 0.338 | 0.647 | 0.389 | 0.044 | 0.049 | 0.502 | 0.077 |
| 4 | LISA (Vanilla) | Only-CLIP | 0.201 | 0.239 | 0.460 | 0.272 | 0.035 | 0.039 | 0.388 | 0.061 |
| 5 | LISA (Vanilla) | Random | 0.122 | 0.159 | 0.287 | 0.174 | 0.028 | 0.034 | 0.261 | 0.046 |
| 6 | LISA (Vanilla) | *GTviews* | *0.293* | *0.340* | *0.645* | *0.389* | *0.046* | *0.051* | *0.510* | *0.079* |
| 7 | LISA-CAD | Cross-Attention | 0.486 | 0.570 | 0.694 | 0.582 | **0.287** | **0.327** | 0.547 | **0.359** |
| 8 | LISA-CAD | FiLM | **0.501** | **0.588** | **0.706** | **0.598** | 0.284 | 0.326 | **0.551** | 0.358 |
| 9 | LISA-CAD | No-Fusion | 0.490 | 0.570 | 0.695 | 0.584 | 0.283 | 0.324 | 0.541 | 0.355 |
| 10 | LISA-CAD | Only-CLIP | 0.305 | 0.374 | 0.464 | 0.377 | 0.160 | 0.193 | 0.327 | 0.210 |
| 11 | LISA-CAD | Random | 0.196 | 0.262 | 0.293 | 0.250 | 0.071 | 0.097 | 0.156 | 0.101 |
| 12 | LISA-CAD | *GTviews* | *0.529* | *0.619* | *0.744* | *0.632* | *0.311* | *0.355* | *0.598* | *0.390* |

**Ablation Studies** We analyze the impact of view ranking formulation and supervision granularity to understand the role of geometric prioritization in language-driven CAD localization. Using the geometric metrics defined earlier, we evaluate segmentation adaptation and view selection design. The metrics re-

**Table 2: Comparison against zero-shot point cloud baselines** on the common 1535-query testset across 218 meshes. Existing point-cloud localization methods achieve high recall but low precision, resulting in poor F1 scores due to over-segmentation.

| | | Face Localization | | | | Edge Localization | | | |
|---|---|---|---|---|---|---|---|---|---|
| **S.no** | **Model** | **IoU** | **Prec.** | **Rec.** | **F1** | **IoU** | **Prec.** | **Rec.** | **F1** |
| 1 | PartSLIP (Default) | 0.073 | 0.083 | 0.368 | 0.114 | 0.012 | 0.013 | 0.422 | 0.022 |
| 2 | PartSLIP (Top-PCD) | 0.062 | 0.081 | 0.250 | 0.097 | 0.012 | 0.013 | 0.262 | 0.021 |
| 3 | Find3D (Default) | 0.171 | 0.172 | **0.955** | 0.260 | 0.017 | 0.017 | **0.937** | 0.032 |
| 4 | Find3D (Top-PCD) | 0.162 | 0.187 | 0.555 | 0.243 | 0.022 | 0.023 | 0.399 | 0.040 |
| 5 | PatchAlign3D (Default) | 0.159 | 0.161 | 0.893 | 0.242 | 0.017 | 0.017 | 0.895 | 0.031 |
| 6 | PatchAlign3D (Top-PCD) | 0.154 | 0.203 | 0.374 | 0.229 | 0.022 | 0.028 | 0.178 | 0.040 |
| 7 | MV-GEL (FiLM@top3, ViewNMS) | 0.372 | 0.400 | 0.846 | 0.495 | 0.180 | 0.194 | 0.695 | 0.266 |
| 8 | MV-GEL (FiLM@top3) | 0.419 | 0.469 | 0.810 | 0.540 | 0.228 | 0.248 | 0.708 | 0.317 |
| 9 | **MV-GEL (FiLM@top1)** | **0.501** | **0.588** | 0.706 | **0.598** | **0.284** | **0.326** | 0.551 | **0.358** |

ported were averaged for all samples in the test set. Results for Face@top1 view and Edge@top1 view are shown in Table 1.

We ablate components of *GELviews*, comparing FiLM and Cross-Attention fusion variants applied in the modality fusion block (Fig.2). The No-Fusion variant bypasses this block by directly using position-aware features for self attention in the view-interaction head, while Only-CLIP removes the view-interaction head and relies solely on the CLIP head. Random view selection is included to assess the net contribution of learned ranking. All view selection strategies are evaluated with both fine-tuned (LISA-CAD) and Vanilla LISA backbones. For a better understanding of the ablated networks, readers are advised to refer to the supplementary.

*Effect of Domain Adaptation* As seen in Table 1, replacing Vanilla LISA with LISA-CAD consistently improves performance across all view selectors for both faces and edges. While Vanilla LISA attains moderate recall, it exhibit's low precision on edges and faces, leading to weak IoU and F1, as shown in Fig. 6(right) . This indicates over-segmentation and limited geometric specificity. CAD fine-tuning substantially improves precision while preserving recall, resulting in markedly stronger F1 and IoU. These results confirm that domain adaptation is critical for geometry-aware segmentation.

*Effect of View Selector Design* Within CAD-LISA, fusion-based view selectors outperform Only-CLIP and Random view selection strategies and perform on-par with No-Fusion. FiLM achieves consistently strong overall performance across top views (for top@3 and top@5 views please refer to the supplementary), closely followed by Cross-Attention and No-Fusion. Cross-Attention yields better metrics for edge localization. Cross-Attention, despite being a powerful multimodal fusion architecture, doesn't outperform FiLM by a great margin. We provide a reason for this performance order in the supplementary. The consistently minor win of fusion-based models over No-Fusion variant indicates that interaction between semantic and view-aware features improves ranking qual-

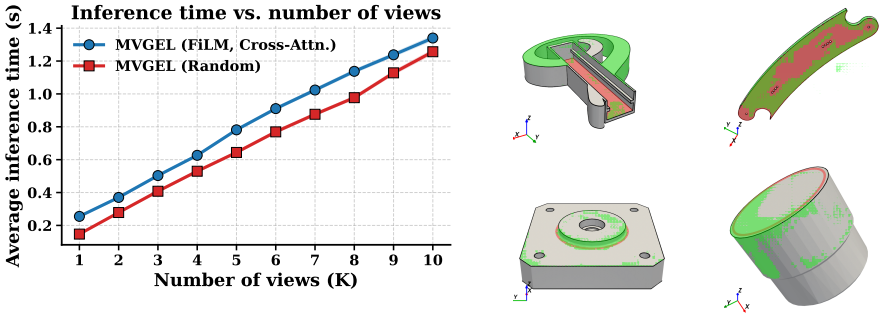



**Fig. 6: Inference efficiency and qualitative localization results. Left:** Inference latency as a function of the number of selected views ($K$). The proposed ranking module adds only $\sim$0.1 s ranking module overhead while achieving comparable face recall using a single selected view, whereas random selection requires at least five views. **Right:** Representative LISA-Vanilla localizations. The top row shows face localization examples and the bottom row shows edge localization examples. Red regions denote ground-truth target segments and green regions denote predicted segments.

ity. The Only-CLIP selector performs significantly worse, particularly for edges, demonstrating that global image-text similarity alone is insufficient for identifying geometrically informative viewpoints. Random view selection yields the lowest performance across both backbones, confirming that gains arise from learned view prioritization. Edge localization is consistently more challenging than face localization due to thin projections, discretization sensitivity, and view-dependent occlusions. Incorrect views can prevent the segmentation backbone from observing the relevant entities described in $q$, leading to missed detections. Nevertheless, relative improvements across selectors are larger for edges, highlighting the importance of view quality and domain-specific segmentation for structurally sensitive features. The combination of CAD-LISA and fusion-based ranking provides the most balanced performance across both entity types.

*Visibility Oracle and View-Selection Headroom* To quantify the absolute effectiveness of our view selection module, we introduce a *GTviews* Ground truth oracle baseline in Table 1. This acts as the theoretical performance ceiling, representing the maximum achievable scores when the target geometric view aligned most with the text prompt is selected. Remarkably, our best-performing FiLM selector achieves a Face F1 score of 0.6, capturing 95% of the oracle's performance limit (0.63 F1). Similarly, for the highly challenging task of edge localization, FiLM reaches an F1 of 0.36, realizing more than 90% of the 0.39 F1 ceiling. This minimal performance gap proves that our prompt-aware view-ranking network operates near-optimally in identifying the best possible viewpoint, bottlenecked only by the inherent segmentation limit's of the reasoning backbone itself. Readers are advised to refer to the supplementary for additional tables with metric analysis for higher top-k values.

*Multi-View Aggregation* Top-K recall curves, as shown in Fig. 5, further validate the ranking quality and view diversity of our proposed modules. The *(GTviews)* trajectory serves as the upper bound for multi-view aggregation. Notably FiLM, Cross-Attention and No-Fusion rapidly close the gap to this oracle within the first 7 to 8 views. The high initial recall demonstrates that the network successfully prioritizes the most geometrically informative views first.

In contrast, the Only-CLIP and Random LISA-CAD baselines start at significantly lower recall and require a large number of views to gradually accumulate surface coverage, proving that purely semantic or arbitrary sampling fails to capture view-dependent geometry efficiently. This advantage is especially pronounced for edge localization (Figure 5b), where geometry-aware fusion dramatically accelerates the coverage of thin, easily occluded entities.

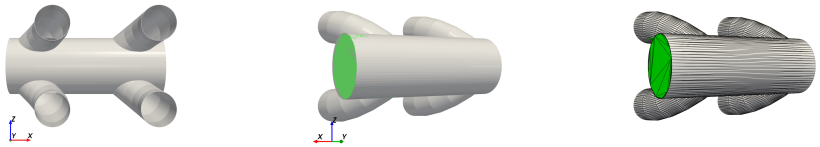
*Baseline Comparisons* In Table 2, we evaluated similar open vocabulary point cloud based segmentation baselines like PartSLIP [32], Find3D [36], PatchAlign3D [13] on our test dataset, by projecting point predictions back to the mesh. We introduced default as well as adapted *Top-PCD* (restricting to top point cloud density predictions matching the caption: 10% for faces, 2% for edges) baselines. Because these models work with point clouds that lack explicit boundary topology and lack deep language reasoning, they severely over-segment ("bleed") across faces (*e.g.*, Find3D achieves 0.95 Recall but only 0.17 Precision) and 1D edges (IoU $\leq 0.022$).

*Inference Cost vs. Views* Fig. 6(Left) explicitly profiles the avg. time/query vs. views. Reaching 0.7 recall on faces (on average on the test set) with 5 random views incurs $> 2\times$ the latency of 1 *GELview* (0.6s for the random views variant vs 0.25s for the FiLM variant). Using **MV-GEL** random takes 16 GB VRAM while MV-GEL (FiLM) takes 17GB for one query on average. Thus *GELviews* yields significantly better metrics at a fraction of the computational cost. The inference was profiled on a single H100 NVIDIA GPU.

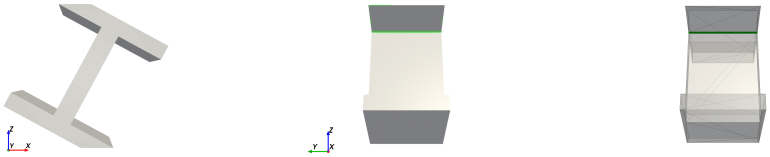
## 7 Conclusions

We introduced **MV-GEL**, a framework for language-driven geometric entity localization that unifies reasoning-based segmentation with prompt-aware view selection. By adapting LISA to the CAD domain and proposing *GELviews*—a geometry-conditioned view ranking module, our method accurately localizes specific target faces and edges without requiring parametric B-Rep data.

Our extensive ablations reveal two critical insights. First, domain-specific fine-tuning is essential; it resolves the severe over-segmentation of thin structural edges seen in open-domain models. Second, our fusion-based view selection dramatically outperforms semantic-only (CLIP) and random sampling. By actively prioritizing geometrically informative viewpoints, our method captures over 90% of the theoretical oracle performance for top-1 face and edge localization, and rapidly accelerates surface coverage during multi-view aggregation.



**Caption**: It's the large flat circular face at one end of the pipe assembly. It's best seen from the right end of the pipe at a level angle.



**Caption**: It's the internal corner edge around the top of the I- structure when seen from the left, looking from underneath.

**Fig. 7: MV-GEL** localization pipeline. (Left) Default mesh view, (Center) *GELviews* selected view (top-1) segmented by LISA-CAD, (Right) MV-GEL localized entity predictions on the mesh.

Ultimately, our results demonstrate that language-driven 3D localization is fundamentally a geometry-aware view prioritization problem, not merely an image-text alignment task. By bridging multimodal reasoning with object understanding from the views, **MV-GEL** establishes an effective solution for geometric entity localization and understanding.

## Acknowledgements

The authors gratefully acknowledge Helmholtz-Zentrum Hereon for providing the GPU computing resources on which all experiments in this work were carried out, and thank Prof. Christian Cyron, Head of the Institute of Material Systems Modeling (MSM), Helmholtz-Zentrum Hereon, for providing the platform and support that made this work possible. The authors also thank their colleagues at MSM for helpful discussions. The authors have no competing interests to declare that are relevant to the content of this article.

## References


1. Abdelreheem, A., Skorokhodov, I., Ovsjanikov, M., Wonka, P.: Satr: Zero-shot semantic segmentation of 3d shapes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15166–15179 (2023)
2. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023)
3. Chen, Q., Wu, C., Ji, J., Ma, Y., Yang, D., Sun, X.: Ipdn: Image-enhanced prompt decoding network for 3d referring expression segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 2132–2140 (2025)
4. Chen, X., Li, Q., Wang, T., Xue, T., Pang, J.: Gennbv: Generalizable next-best-view policy for active 3d reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16436–16445 (2024)
5. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al.: Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24185–24198 (2024)
6. Connolly, C.: The determination of next best views. In: Proceedings. 1985 IEEE international conference on robotics and automation. vol. 2, pp. 432–435. IEEE (1985)
7. Dai, W., Li, J., Li, D., Tiong, A., Zhao, J., Wang, W., Li, B., Fung, P.N., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning. Advances in neural information processing systems **36**, 49250–49267 (2023)
8. Ding, H., Liu, C., Wang, S., Jiang, X.: Vision-language transformer and query generation for referring segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 16321–16330 (2021)
9. Garosi, M., Tedoldi, R., Boscaini, D., Mancini, M., Sebe, N., Poiesi, F.: 3d part segmentation via geometric aggregation of 2d visual features. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 3257–3267. IEEE (2025)
10. Goyal, A., Law, H., Liu, B., Newell, A., Deng, J.: Revisiting point cloud shape classification with a simple and effective baseline. In: International conference on machine learning. pp. 3809–3820. PMLR (2021)
11. Guo, Z., Zhang, R., Zhu, X., Tang, Y., Ma, X., Han, J., Chen, K., Gao, P., Li, X., Li, H., et al.: Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. arXiv preprint arXiv:2309.00615 (2023)
12. Guo, Z., Tang, Y., Zhang, R., Wang, D., Wang, Z., Zhao, B., Li, X.: Viewrefer: Grasp the multi-view knowledge for 3d visual grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15372–15383 (2023)
13. Hadgi, S., Gong, B., Sundararaman, R., Pierson, E., Li, L., Wonka, P., Ovsjanikov, M.: Patchalign3d: Local feature alignment for dense 3d shape understanding. arXiv preprint arXiv:2601.02457 (2026)
14. He, S., Jie, G., Wang, C., Zhou, Y., Hu, S., Li, G., Ding, H.: Refersplat: Referring segmentation in 3d gaussian splatting. arXiv preprint arXiv:2508.08252 (2025)
15. Jatavallabhula, K.M., Kuwajerwala, A., Gu, Q., Omama, M., Chen, T., Maalouf, A., Li, S., Iyer, G., Saryazdi, S., Keetha, N., et al.: Conceptfusion: Open-set multimodal 3d mapping. arXiv preprint arXiv:2302.07241 (2023)
16. Jayaraman, P.K., Lambourne, J.G., Desai, N., Willis, K.D., Sanghi, A., Morris, N.J.: Solidgen: An autoregressive model for direct b-rep synthesis. arXiv preprint arXiv:2203.13944 (2022)

17. Jayaraman, P.K., Sanghi, A., Lambourne, J.G., Willis, K.D., Davies, T., Shayani, H., Morris, N.: Uv-net: Learning from boundary representations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11703–11712 (2021)
18. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International conference on machine learning. pp. 4904–4916. PMLR (2021)
19. Jones, B., Hildreth, D., Chen, D., Baran, I., Kim, V.G., Schulz, A.: Automate: A dataset and learning approach for automatic mating of cad assemblies. ACM Transactions on Graphics (TOG) **40**(6), 1–18 (2021)
20. Kareem, A., Lahoud, J., Cholakkal, H.: Paris3d: Reasoning-based 3d part segmentation using large multimodal model. In: European Conference on Computer Vision. pp. 466–482. Springer (2024)
21. Kerr, J., Kim, C.M., Goldberg, K., Kanazawa, A., Tancik, M.: Lerf: Language embedded radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19729–19739 (2023)
22. Khan, M.S., Sinha, S., Sheikh, T.U., Stricker, D., Ali, S.A., Afzal, M.Z.: Text2cad: Generating sequential cad designs from beginner-to-expert level text prompts. Advances in Neural Information Processing Systems **37**, 7552–7579 (2024)
23. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4015–4026 (2023)
24. Koch, S., Matveev, A., Jiang, Z., Williams, F., Artemov, A., Burnaev, E., Alexa, M., Zorin, D., Panozzo, D.: Abc: A big cad model dataset for geometric deep learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9601–9611 (2019)
25. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: Lisa: Reasoning segmentation via large language model. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9579–9589 (2024)
26. Lambourne, J.G., Willis, K.D., Jayaraman, P.K., Sanghi, A., Meltzer, P., Shayani, H.: Brepnet: A topological message passing system for solid models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12773–12782 (2021)
27. Li, J., Ma, W., Li, X., Lou, Y., Zhou, G., Zhou, X.: Cad-llama: leveraging large language models for computer-aided design parametric 3d model generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18563–18573 (2025)
28. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International conference on machine learning. pp. 19730–19742. PMLR (2023)
29. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10965–10975 (2022)
30. Liu, C., Ding, H., Jiang, X.: Gres: Generalized referring expression segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 23592–23601 (2023)
31. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in neural information processing systems **36**, 34892–34916 (2023)

32. Liu, M., Zhu, Y., Cai, H., Han, S., Ling, Z., Porikli, F., Su, H.: Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21736–21746 (2023)
33. Liu, Y., Shekhar Dutt, N., Li, C., Mitra, N.J.: B-repler: Semantic b-rep latent editor using large language models. arXiv e-prints pp. arXiv–2508 (2025)
34. Lu, J., Batra, D., Parikh, D., Lee, S.: Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. Advances in neural information processing systems **32** (2019)
35. Luo, T., Rockwell, C., Lee, H., Johnson, J.: Scalable 3d captioning with pretrained models. Advances in Neural Information Processing Systems **36**, 75307–75337 (2023)
36. Ma, Z., Yue, Y., Gkioxari, G.: Find any part in 3d. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7818–7827 (2025)
37. Mei, G., Riz, L., Wang, Y., Poiesi, F.: Geometrically-driven aggregation for zero-shot 3d point cloud understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 27896–27905 (2024)
38. Peng, S., Genova, K., Jiang, C., Tagliasacchi, A., Pollefeys, M., Funkhouser, T., et al.: Openscene: 3d scene understanding with open vocabularies. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 815–824 (2023)
39. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: Visual reasoning with a general conditioning layer. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
40. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
41. Ren, Z., Huang, Z., Wei, Y., Zhao, Y., Fu, D., Feng, J., Jin, X.: Pixellm: Pixel reasoning with large multimodal model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 26374–26383 (2024)
42. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 945–953 (2015)
43. Team, G., Georgiev, P., Lei, V.I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S., et al.: Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv preprint arXiv:2403.05530 (2024)
44. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
45. Wang, C., Pelillo, M., Siddiqi, K.: Dominant set clustering and pooling for multi-view 3d object recognition. arXiv preprint arXiv:1906.01592 (2019)
46. Wang, Z., Lu, Y., Li, Q., Tao, X., Guo, Y., Gong, M., Liu, T.: Cris: Clip-driven referring image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11686–11695 (2022)
47. Willis, K.D., Pu, Y., Luo, J., Chu, H., Du, T., Lambourne, J.G., Solar-Lezama, A., Matusik, W.: Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. ACM Transactions on Graphics (TOG) **40**(4), 1–24 (2021)

48. Xia, Z., Han, D., Han, Y., Pan, X., Song, S., Huang, G.: Gsva: Generalized segmentation via multimodal large language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3858–3869 (2024)
49. Xu, J., Wang, C., Zhao, Z., Liu, W., Ma, Y., Gao, S.: Cad-mllm: Unifying multimodality-conditioned cad generation with mllm. arXiv preprint arXiv:2411.04954 (2024)
50. Yu, L., Lin, Z., Shen, X., Yang, J., Lu, X., Bansal, M., Berg, T.L.: Mattnet: Modular attention network for referring expression comprehension. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1307–1315 (2018)
51. Yuan, Y., Li, W., Liu, J., Tang, D., Luo, X., Qin, C., Zhang, L., Zhu, J.: Osprey: Pixel understanding with visual instruction tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28202–28211 (2024)
52. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: Pointclip: Point cloud understanding by clip. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8552–8562 (2022)
53. Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., Tang, M., Wang, J.: Fast segment anything. arXiv preprint arXiv:2306.12156 (2023)
54. Zhou, Y., Gu, J., Li, X., Liu, M., Fang, Y., Su, H.: Partslip++: Enhancing low-shot 3d part segmentation via multi-view instance segmentation and maximum likelihood estimation. arXiv preprint arXiv:2312.03015 (2023)
55. Zhu, X., Zhang, R., He, B., Guo, Z., Zeng, Z., Qin, Z., Zhang, S., Gao, P.: Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2639–2650 (2023)
56. Zou, X., Yang, J., Zhang, H., Li, F., Li, L., Wang, J., Wang, L., Gao, J., Lee, Y.J.: Segment everything everywhere all at once. Advances in neural information processing systems **36**, 19769–19782 (2023)