What One View Reveals, Another Conceals: 3D-Consistent Visual Reasoning with LLMs

Anonymous authors
Paper under double-blind review

Abstract

Maintaining semantic label consistency across multiple views is a persistent challenge in 3D semantic object detection. Existing zero-shot approaches that combine 2D detections with vision-language features often suffer from bias toward non-descriptive viewpoints and require a fixed label list to operate on. We propose a truly open-vocabulary algorithm that uses large language model (LLM) reasoning to relabel multi-view detections, mitigating errors from poor, ambiguous viewpoints and occlusions. Our method actively samples informative views based on feature diversity and uncertainty, generates new label hypotheses via LLM reasoning, and recomputes confidences to build a spatial-semantic representation of objects. Experiments on controlled single-object and multi-object scenes show double digit improvement, in accuracy and sampling rate over ubiquitous fusion methods using YOLO, and CLIP. We demonstrate in multiple cases that **L**LM-guided **A**ctive **D**etection and **R**easoning (LADR) balances detail preservation with reduced ambiguity and low sampling rate. We provide theoretical convergence analysis showing exponential convergence to a stable and correct semantic label.

1 Introduction

Consistently detecting objects across multiple viewpoints is a crucial task for autonomous agents, such as drones and robots. A single object may appear vastly different depending on the viewpoint, lighting, or degree of occlusion, and visual features extracted from such views often drift in embedding space. As a result, inconsistent labels emerge when fusing detections across views, leading to degraded spatial-semantic representations and downstream performance.

Recent zero-shot approaches (Jatavallabhula et al., 2023; Peng et al., 2023; Cartillier et al., 2021), address this by combining off-the-shelf detectors (Redmon et al., 2016) with vision-language models (Radford et al., 2021; Cherti et al., 2023) to assign open-vocabulary labels in 3D. While these methods avoid task-specific retraining, they rely heavily on two components: (1) the accuracy of the underlying detector, and (2) the similarity between extracted image features and a user-defined list of candidate labels. Both dependencies introduce bottlenecks. First, misdetections or low-quality views (such as those from the back of an object) can dominate the fused feature representation, biasing the final label. Second, reliance on a user-defined list of labels limits true open-vocabulary capability, hampers generalization to novel categories, and constrains the level of detail that can be captured for each object.

We propose a different approach referred to as **LADR** (**L**LM-guided **A**ctive **D**etection and **R**easoning). LADR uses large language model (LLM) reasoning to actively refine and reweight multi-view detections. Instead of passively aggregating features, our method iteratively samples informative viewpoints based on feature diversity, prompts an LLM to generate and refine label hypotheses from available visual evidence, and recomputes label confidences accordingly. This reasoning process reduces the influence of misleading views, removes the need for a fixed label set, and enables a more robust spatial-semantic representation of the scene. We provide rigorous Markov process-based analysis for an exponential convergence rate to consistent labels. It shows differentiation in rates on the ablated versions of LADR, proving that active uncertainty based sampling, with geometric grounding is the best approach among LADR algorithms.

Our contributions are as follows:

- **LLM-guided relabeling for 3D consistency:** An open-vocabulary method that uses LLM reasoning to correct viewpoint-induced misclassifications without retraining.
- **Smart sampling strategy:** An active selection of views based on feature diversity, uncertainty estimation, and geometric grounding, balancing detail preservation with reduced context ambiguity.
- **Spatial-semantic mapping:** A representation that integrates refined labels with object geometry, suitable for downstream 3D tasks.
- **Comprehensive evaluation:** single-object and multi-object experiments across diverse environments, showing improvement of over 40%, respectively, in 3D semantic label accuracy and sampling rate, over ubiquitous fusion methods using YOLO, and CLIP.
- **Theoretical analysis** for each ablated LADR version, proving exponential convergence to consistent semantic labels, with increasingly stronger constants for added components.

Our contributions establish a framework for zero-shot open-vocabulary 3D understanding that combines semantic reasoning, efficient view selection, and spatial integration, leading to more robust and consistent labeling across diverse scenarios.

2 Related work

Foundation Models in Object Detection. Object detection has rapidly advanced from region-based CNNs and single-stage detectors to foundation models, which enable more general and flexible representations beyond closed-set training. Architectures such as YOLO-World and YOLOE (Cheng et al., 2024; Wang et al., 2025) leverage large-scale pretraining to improve detection accuracy and adaptability across diverse scenarios. Vision-language models (VLMs) like CLIP (Radford et al., 2021; Cherti et al., 2023) provide open-vocabulary capabilities by connecting visual features with text embeddings, while models such as Segment Anything (Kirillov et al., 2023) offer class-agnostic segmentation that can be integrated into detection pipelines. Multimodal large language models like GPT-4V (OpenAI, 2024) further complement these approaches by enabling zero-shot reasoning over visual inputs, making them useful for refining labels and guiding exploration. These approaches demonstrate the potential to reduce reliance on task-specific training and expand detection to previously unseen categories.

Open-Vocabulary 3D Object Detection. ConceptFusion (Jatavallabhula et al., 2023) builds open-vocabulary 3D object maps by combining pretrained VLMs with 3D scene representations. The method uses YOLO-World as an initial object detector and Segment Anything for segmentation, attaching VLM features (e.g., from CLIP) to 3D points reconstructed from RGB-D scans, with features from multiple 2D observations aggregated via simple averaging (which ignores the 3D consistency problem). While it aims to assign open-vocabulary labels, the object categories are ultimately constrained to a fixed set. Peng et al. (2023) takes a voxel-based approach, backprojecting per-pixel CLIP features into a 3D voxel grid and fusing multiple views using different pooling strategies (random, median, or mean) among these approaches, mean pooling yields the most stable results. Kassab et al. (2024) revisits design choices for open-vocabulary 3D labeling by selecting a single "best" view per object based on a confidence metric, with the entropy of CLIP similarities with category embeddings performing best. In contrast, LADR leverages LLM reasoning to iteratively identify and reweight informative views, producing a more robust spatial-semantic representation that is less sensitive to viewpoint bias and not limited by a fixed label set.

Active exploration. Active exploration in embodied agents aims to optimize camera or agent trajectories to reduce uncertainty and collect informative observations. SEAL Chaplot et al. (2021) and subsequent works Scarpellini et al. (2024) introduce a self-supervised framework in which agents explore their environment to learn semantic segmentation without manual labels, leveraging 3D spatial consistency. These methods train an exploration policy to target novel or uncertain areas, optimizing coverage of diverse object views. Features from multiple viewpoints are reprojected into a shared 3D space using depth and camera

poses, and a 3D consistency loss ensures that features corresponding to the same physical point remain consistent across views. This supervision enables learning of a semantic segmentation function directly from RGB-D frames, without human annotations, and replaces random or fixed path planning with informed, targeted exploration. While effective, these approaches often require reinforcement learning policies and multiple rollouts, which can be computationally expensive. In contrast, zero-shot LLM-based methods can reason about object semantics and its correlation with the viewpoint directly from observations without task-specific policy training, avoiding the overhead and sample inefficiency inherent to learned exploration policies.

3 The 3D Consistency Problem

Achieving consistent object labeling across multiple viewpoints remains a key obstacle in 3D perception. In multi-view pipelines, each observation of an object is processed independently before being fused into a unified label. When these observations are heterogeneous (due to varying viewpoints, occlusions, or lighting) the resulting feature embeddings can drift toward non-representative appearances. This drift can overweight misleading views, leading to label instability.

In zero-shot approaches such as those combining YOLO detections with CLIP embeddings, the problem is exacerbated by two factors:

1. **Viewpoint sensitivity:** Descriptive views (e.g., the front of a piano) and non-descriptive views (e.g., the back of the same piano) contribute equally to the aggregated embedding. If the majority of views lack discriminative features, the resulting label can shift toward incorrect categories.

2. **Label space constraints:** Even in open-vocabulary settings, relying on a fixed set of candidate labels constrains the level of detail that can be captured for each object, e.g., labeling a chair simply as 'furniture' rather than distinguishing it as an 'office swivel chair.'

To illustrate the severity of this issue, we consider a controlled example where images are taken around a piano. We define **good views** as those from the front, containing distinctive features, and **bad views** as those from the back, lacking such cues. In a progressive experiment, we start with three good views and incrementally replace them with bad ones, testing multiple labeling strategies. The task is to assign a single label to the object, given all current views.

Method	3 Good / 0 Bad	2 Good / 1 Bad	1 Good / 2 Bad
YOLOE Constrained	**piano (0.25)**	**piano (0.21)**	crate (0.26)
YOLOE ScanNet200	cabinet (0.78)	cabinet (0.61)	cabinet (0.51)
YOLOE RAM	chiffonier (0.90)	wall (0.16)	wall (0.17)
CLIP Constrained	**piano (0.31)**	**piano (0.27)**	crate (0.26)
CLIP ScanNet200	**piano (0.31)**	**piano (0.27)**	crate (0.26)
CLIP RAM	**piano (0.31)**	**piano (0.27)**	oak (0.27)
LLM	**acoustic piano**	**acoustic piano**	**acoustic piano**

Table 1: **Piano viewpoint bias experiment.** "Good" images show the piano front, while "Bad" images show the back. Each cell reports the *predicted label (confidence)*, with correct predictions shown in **bold**. For YOLOE baselines, the most frequent label is selected, whereas CLIP baselines choose the label with the highest similarity. The "Constrained" setting restricts candidate labels to "piano" and "crate," while "ScanNet200" (Rozenberszki et al., 2022) and "RAM" (Recognize Anything Model class list of over four thousand categories, (Zhang et al., 2023)) use their respective class lists to select the most probable label. The LLM is prompted to give a more specific label than a simple class label."

As shown in Table 1, methods relying solely on YOLO or CLIP degrade quickly as bad views increase. In the 1-good / 2-bad case, CLIP-based methods incorrectly label the piano as "crate" or "oak," while YOLO struggles particularly when the label space is large, producing highly inconsistent predictions. In contrast, the LLM-based approach consistently selects the correct and more detailed label across all conditions. However, the LLM does not provide calibrated confidence values, making it difficult to assess the reliability of its predictions. This observation motivates LADR's hybrid strategy: combining the reasoning capabilities of LLMs with the quantitative confidence scores from CLIP. Our approach allows for both robust label selection and informed weighting across views, mitigating the effects of viewpoint bias and constrained label spaces.

4 Notation and Workflow

We consider multi-view object labeling in 3D scenes. For simplicity, and without loss of generality, we address a single object. Let $\mathcal{I} = \{I_1, \ldots, I_N\}$ denote the initial set of RGB-D images captured around a target object, where N is the number of views. Each image I_i is accompanied by depth information D_i and camera pose P_i. For each image, object observations are extracted using a combination of a detector, a feature extractor and a segmentation model as

$$O_i = \text{DetectAndSegment}(I_i, D_i, P_i),$$

and merged across all views into a spatial-semantic map

$$\mathcal{M} = \text{MergeObservations}(\{O_1, \ldots, O_N\}),$$

which accumulates object points, labels, and features into a coherent 3D representation, analogous to ConceptFusion's fusion (Jatavallabhula et al., 2023). We define a function for relabeling as

$$\mathcal{M}_{\text{refined}},\ P_{\text{next}} = \text{RefineAndPropose}(\mathcal{M}, \mathcal{I}),$$

which applies LLM reasoning to refine labels in $\mathcal{M}$ and selects the most informative next viewpoint.

Our workflow proceeds iteratively: images are captured and objects inside it are merged into $\mathcal{M}$, refined, and used to propose the next viewpoint. This repeats until labels reach sufficient confidence or a maximum number of views is obtained.

5 Methodology

In this section, we present our algorithm for LLM-guided multi-view object labeling. In multi-view labeling, the evolving set of detection images at each iteration often contains a mix of highly informative canonical views, ambiguous and redundant observations. Presenting all available images to the LLM simultaneously is problematic: it risks pushing the model toward a generic, lowest-common-denominator label, substantially increases computational cost, and may even exceed the LLM's context window. A possible workaround is to tile multiple views into a single composite image, but this forces downsampling that discards fine-grained details. To address these challenges, we adopt an iterative inner loop that samples a small subset of images to form a hypothesis and then prunes away views that conflict with it.

We introduce two ablated LADR studies prior to presenting our algorithm, to facilitate the introduction of LADR. We focus on the RefineAndPropose function which defines each algorithm. We also provide convergence analysis for each algorithm, showing improved convergence rates with the addition of key algorithmic components.

5.1 LLM-Random: Basic Hypothesis Proposal and Killing

The first ablated version, **LLM-Random**, introduces the fundamental hypothesis-proposal and iterative image removal procedure. The LLM-Random variant implements this process using the simplest possible sampling strategy: uniform random selection. The workflow of the algorithm is illustrated in Figure **??**, and its pseudo code in Alg. 1.

Algorithm 2 Given a set of detection images and camera poses, the algorithm iteratively samples views and queries a large language model (LLM) to propose a hypothesis $\hat{y}_t$ together with a boolean confidence indicator c_t. The confidence indicator denotes whether the LLM considers the current hypothesis sufficiently reliable to terminate the procedure; otherwise, an uninformative view is pruned and the process continues until a confident hypothesis is obtained or a maximum number of iterations is reached.

Require: Detection images $\mathcal{I}$, camera angles $\mathcal{P}$, sample size N
$\mathcal{M}_{\text{refined}} \leftarrow \varnothing$, $t \leftarrow 0$
while $|\mathcal{I}| \geq N$ **and** $t < T_{\max}$ **do**
  $t \leftarrow t + 1$
  $\mathcal{I}_{\text{sample}} \leftarrow \text{RandomSample}(\mathcal{I}, N)$
  $(\hat{y}, \mathrm{c}_t, I_{\text{kill}}, P_{\text{next}}) \leftarrow \text{LLM\_Query}(\mathcal{I}_{\text{sample}}, \mathcal{P})$
  if $\mathrm{c}_t = \text{True}$ **then**
    $\mathcal{M}_{\text{refined}} \leftarrow \hat{y}$
    break
  end if
  $\mathcal{I} \leftarrow \mathcal{I} \setminus \{I_{\text{kill}}\}$
  $\mathcal{M}_{\text{refined}} \leftarrow \hat{y}$
end while
return $(\mathcal{M}_{\text{refined}}, P_{\text{next}})$

The process repeats until either (1) the LLM reports confidence in its label, or (2) the detection set $\mathcal{I}$ has been reduced to fewer than N images (3) a maximum number of iterations is reached; in which case the algorithm returns the refined map $\mathcal{M}_{\text{refined}}$ along with the next proposed viewpoint P_{next}. The LLM prompt used for this algorithm is provided in Appendix B.7.

5.1.1 Theoretical analysis

For an apples-to-apples comparison with methods that compare two views per iteration, we analyze the randomized baseline in the minimal $N = 2$ setting. The extension to general N follows by replacing pairwise exposure and dominance constants with their N-subset counterparts.

Theorem 1 (High-probability elimination of bad views under LLM-kill random-pair pruning) *Let $\mathcal{I}_0$ be a finite set of views, and let $G \subseteq \mathcal{I}_0$ and $B = \mathcal{I}_0 \setminus G$ denote the good and bad views, with $G \neq \varnothing$. At iteration t, let $\mathcal{I}_t$ be the retained set and define*

$$G_t := G \cap \mathcal{I}_t, \qquad B_t := B \cap \mathcal{I}_t, \qquad b_t := |B_t|.$$

Assume the following hold for all t:

(A1) Exposure under random sampling. *There exists $\delta > 0$ such that whenever $b_t > 0$,*

$$\Pr(\mathcal{I}_{\text{sample},t} \cap B_t \neq \varnothing \mid \mathcal{F}_t) \geq \delta, \quad \mathcal{F}_t := \sigma(\{X_s : s \leq t\}).$$

(A2) LLM kill-dominance on mixed pairs. *There exists $\eta_{\text{kill}} > \frac{1}{2}$ such that whenever $\mathcal{I}_{\text{sample},t} = \{I_g, I_b\}$ with $I_g \in G_t$ and $I_b \in B_t$,*

$$\Pr(I_{\text{kill},t} = I_b \mid \mathcal{I}_{\text{sample},t} = \{I_g, I_b\},\ \mathcal{F}_t) \geq \eta_{\text{kill}}.$$

Define $\beta := \delta\, \eta_{\text{kill}} \in (0, 1)$ and let $b_0 := |B|$.

$$\text{for any } T \geq 1, \quad \Pr(b_T > 0) \ \leq\ \Pr\big(\text{Bin}(T, \beta) < b_0\big). \text{ In particular, if } T \geq \frac{2b_0}{\beta} \text{ then}$$

$$\Pr[b_T > 0] \ \leq\ \exp\left(-\frac{\beta T}{8}\right) \ =\ \exp(-c_2 T), \qquad c_2 := \frac{\beta}{8}.$$

Equivalently, for any $\varepsilon \in (0, 1)$ it suffices to take $T \ \geq\ \max\left\{\frac{2b_0}{\beta},\ \frac{8}{\beta}\log\frac{1}{\varepsilon}\right\}$ to guarantee $\Pr[b_T > 0] \leq \varepsilon$.

5.2 LLM-Sampling: CLIP-Guided Selection and Confidence

The **LLM-Sampling** algorithm follows similar structure as LLM-Random, but improves upon it by leveraging image embeddings provided by a contrastive VLM (eg. CLIP, Cherti et al. (2023)) for both image selection and confidence assessment; we refer to these embeddings as *CLIP features*. We illustrate the method using two sampled images per iteration for clarity. In practice, this generalizes to two subsets of images, sampled similarly. We provide a sketch of an iteration in Figure 1, and pseudo code in Algorightm 3.

Sampling. Instead of randomly sampling images, this algorithm identifies two images $I_{\text{rep}}, I_{\text{amb}} \subset \mathcal{I}$ based on their cosine similarity of CLIP features relative to the current label hypothesis: the closest (most representative) and the farthest (potentially ambiguous) image. The initial hypothesis can be set using the most common detection label (e.g. YOLO detections). I_{rep} and I_{amb} are then fed to the LLM to generate a new label hypothesis $\mathcal{M}_{\text{refined}}$ and propose the next best view P_{next}. This procedure balances *exploitation* (focusing on the most representative view) with *exploration* (including a diverse, informative view).

Confidence Computation and Removal. A global object representation is computed by averaging CLIP features across all current images. Cosine similarity between this global feature and the LLM label embedding $\mathcal{M}_{\text{refined}}$ provides a confidence score for the proposed label. Similarities are computed between $\hat{y}{=}\mathcal{M}_{\text{refined}}$ and the sampled detections I_{rep}, I_{amb}. The less similar detection is discarded from $\mathcal{I}_t$. To determine whether the current label is reliable enough to be accepted or if further iterations with new images are required, a confidence threshold is applied (see Appendix B for its calibration).

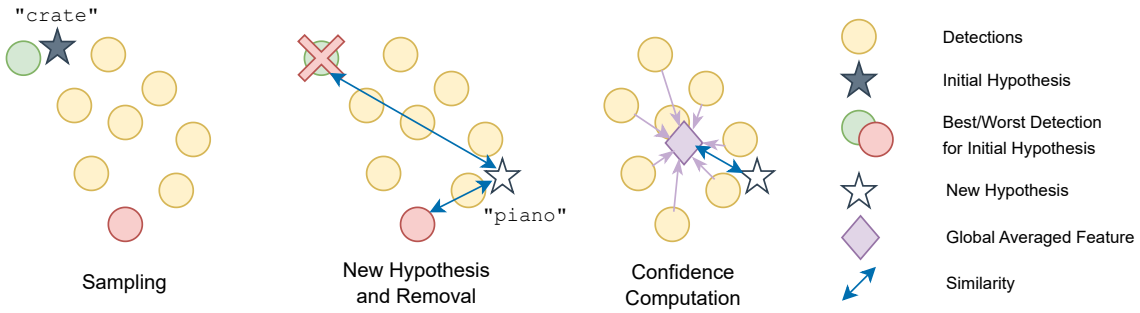

Figure 1: **Visualization of the LLM-Sampling algorithm**: (left) two images are selected based on feature distance from current hypothesis, (middle) a new label hypothesis is generated from the two images, and the less similar detection is removed, (right) a global averaged feature and global confidence are computed.

Recovering the Final Hypothesis via Caching. In some cases, the LLM may generate an accurate label early on, but it cannot yet be accepted due to insufficient supporting evidence. The CLIP-similarity-based confidence computation allows for the re-evaluation of previously generated hypotheses. When new images are introduced or when the hypothesis-proposal loop concludes, the most confident hypothesis is retrieved from the cache. Empirically, this mechanism reduces noise from LLM hallucinations, prevents sudden label shifts, and improves convergence consistency.

Advantages. By selecting views using CLIP feature distances, our approach presents the LLM with more informative and diverse samples than random subset selection, reducing redundancy and improving sample efficiency. CLIP-based similarity scores also provide a more stable and interpretable confidence signal than the LLM's self-reported confidence used in LLM-Random. In addition, hypothesis caching allows candidate labels to be retained and re-evaluated without repeated LLM calls, improving both efficiency and robustness. Together, these strategies effectively combine the generative strengths of LLMs with the contrastive structure of CLIP models, leading to more reliable and faster convergence.

Algorithm 4 The algorithm iteratively selects representative and ambiguous views using CLIP features, queries an LLM to propose a label, and prunes inconsistent images based on feature-level agreement, retaining the hypothesis with the highest global consistency score.

Require: Detection images $\mathcal{I}$, camera angles $\mathcal{P}$, sample size $N = 2$
Initialize hypothesis cache $\mathcal{C} \leftarrow \varnothing$
$t \leftarrow 0$
while $|\mathcal{I}| \geq N$ and $t < T_{\max}$ **do**
  $t \leftarrow t + 1$
  Compute global representation: $\mathbf{g} \leftarrow \frac{1}{|\mathcal{I}|} \sum_{I \in \mathcal{I}} f_{\text{CLIP}}(I)$
  Select informative views: $\forall \mathcal{I}$: compute $s_I = \cos(f_{\text{CLIP}}(I), \mathbf{g})$
  $I_{\text{rep}} \leftarrow \arg\max_{I \in \mathcal{I}} s_I$ ▷ closest to mean, most representative
  $I_{\text{amb}} \leftarrow \arg\min_{I \in \mathcal{I}} s_I$ ▷ farthest from mean, most ambiguous
  $\mathcal{I}_{\text{sample}} \leftarrow \{I_{\text{rep}}, I_{\text{amb}}\}$
  Query LLM to propose a label: $(\hat{y}, h, P_{\text{next}}) \leftarrow \text{LLM\_Query}(\mathcal{I}_{\text{sample}}, \mathcal{P})$
  Evaluate hypothesis confidence: $c_{\text{global}} \leftarrow \cos(h, \mathbf{g})$
  Determine which image harms consistency more:
  if $s_{amb} < s_{rep}$ **then**: Remove I_{amb} from $\mathcal{I}$
  else: Remove I_{rep} from $\mathcal{I}$
  end if
  Cache hypothesis: Store $(\hat{y}, h, c_{\text{global}})$ in $\mathcal{C}$
end while
Recover most confident hypothesis: $(y^*, h^*, c^*) \leftarrow \arg\max_c (\hat{y}, h, c) \in \mathcal{C}$
return (y^*, P_{next})

5.2.1 Theoretical analysis

Theorem 2 (Exponential elimination of bad views under hypothesis-driven extreme selection) *Let $\mathcal{I}_0$ be a finite set of views of an object, and let $G \subseteq \mathcal{I}_0$ and $B = \mathcal{I}_0 \setminus G$ denote the sets of* good *and* bad *views, where good views provide reliable evidence of the true canonical label y^* and bad views are ambiguous. Assume $G \neq \varnothing$. At iteration t, the algorithm maintains a retained set $\mathcal{I}_t \subseteq \mathcal{I}_0$ and a hypothesis h_t.*

Define: $$G_t = G \cap \mathcal{I}_t, \qquad B_t = B \cap \mathcal{I}_t, \qquad b_t = |B_t|.$$

Assume:

(C1) Extreme selection exposes good and bad views. *There exists $\delta > 0$ such that, whenever $b_t > 0$,*

$$\Pr\big(I_{\text{amb},t} \in B_t,\ I_{\text{rep},t} \in G_t \mid \mathcal{F}_t\big) \geq \delta, \quad \mathcal{F}_t := \sigma(\{X_s : s \leq t\}).$$

(C2) Good-vs-bad semantic dominance. *There exists $\eta > 1/2$ such that, whenever the selected unordered pair $\{I_{\text{rep},t}, I_{\text{amb},t}\}$ consists of exactly one good view $I_g \in G_t$ and one bad view $I_b \in B_t$, the refined hypothesis h_{t+1} produced by the LLM satisfies*

$$\Pr\Big(\cos\big(h_{t+1}, f_{\text{CLIP}}(I_g)\big) > \cos\big(h_{t+1}, f_{\text{CLIP}}(I_b)\big) \,\Big|\, \{I_{\text{rep},t}, I_{\text{amb},t}\} = \{I_g, I_b\}, \mathcal{F}_t\Big) \ \geq\ \eta.$$

Then there exist $c_1, c_2 > 0$ (depending only on δ and η) such that for all $T \geq 0$, $\Pr[b_T > 0] \ \leq\ c_1 e^{-c_2 T}$.

Namely, the probability that at least one bad view remains after T iterations decays exponentially in T. Or, for any $\varepsilon \in (0, 1)$, all bad views are eliminated after $O(\log(1/\varepsilon))$ iterations with probability at least $1 - \varepsilon$.

5.3 LLM-Polygon: Spatially Grounded Refinement

The LADR algorithms presentation, **LLM-Polygon**, extends LLM-Sampling by incorporating spatial grounding into the label refinement process. This addition allows the algorithm to reason about coverage of the object's geometry, to guide exploration, and to prioritize views that reduce semantic uncertainty, see Figure 2. We provide the pseudo code for LLM-Polygon in Algorithm 5.

Spatial Assignment. A right-prism polygon (or an icosahedron) is constructed around the object to approximate its spatial extent. Each detection is associated with the polygon faces it observes, determined by projecting camera rays on the polygon faces. This partition grounds the detections into spatially meaningful subsets and prevents over-representation of individual sides.

Per-Face Confidence. For each polygon face, CLIP features of the associated detections are averaged to form a local feature representation. Unobserved faces are assigned an *uncertainty weight*, a hyperparameter that trades off exploration and exploitation: lower uncertainty weights promote taking additional views, while higher values enable faster convergence by downweighting unseen sides (see Appendix B for a calibration guide). Global confidence is then computed as the average similarity between the current label hypothesis and the per-face features of existing images in I_t.

Iterative Refinement. Label proposal and image pruning proceed as in LLM-Sampling, but the next viewpoint is chosen using spatial confidence and coverage, and not via an LLM. Specifically, P_{next} is selected as the face whose neighboring faces exhibit the largest confidence difference, with priority given to previously unseen sides. This active mechanism directs exploration toward underrepresented object regions.

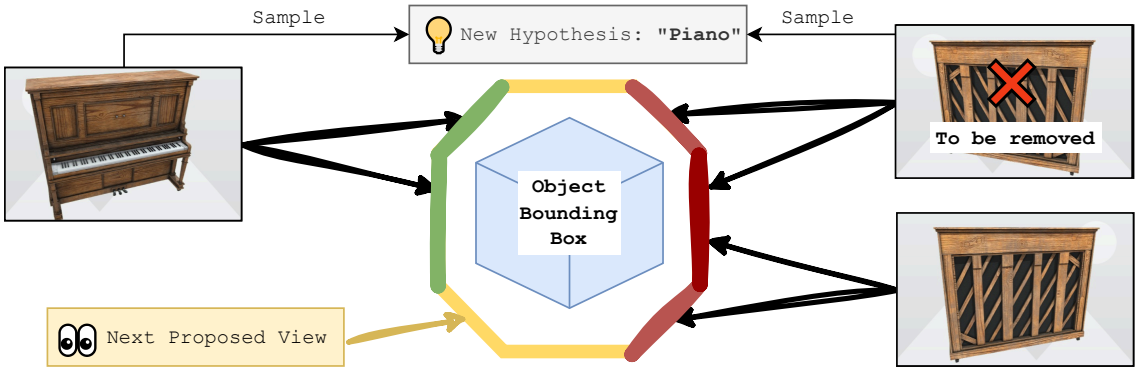

Figure 2: **Illustration of LLM-Polygon**: Object detections are spatially grounded to polygon faces. Per-face confidence is computed based on CLIP features: green sides correspond to high visual similarity to the current label ("piano"), red sides indicate low similarity, and yellow sides represent unseen faces. The next camera viewpoint is selected to reduce uncertainty, prioritizing unseen faces.

5.4 Theoretical Analysis

LLM-Polygon introduces three mechanisms absent from LLM-Sampling.

(A) Spatial partitioning via polygon faces. Each view is assigned to one or more polygon faces $F \in \mathcal{F}$, based on camera-ray intersection. This ensures balanced geometric coverage and eliminates oversampling of a single side of the object.

Theoretical implication: Every side of the object is guaranteed to be observed, so *good views always have nonzero sampling probability.* In LLM-Sampling this had to be assumed; here it is built into the algorithm.

(B) Per-face feature averaging with uncertainty weighting. Each face F computes a representative embedding

$$\mathbf{f}_F = \begin{cases} \frac{1}{|\mathcal{I}_F|} \sum_{I \in \mathcal{I}_F} f_{\text{CLIP}}(I), & \mathcal{I}_F \neq \varnothing, \\ \lambda \mathbf{u}, & \mathcal{I}_F = \varnothing, \end{cases}$$

where $\mathbf{u}$ is a calibrated uncertainty vector. Then each image receives a similarity score $s_I = \cos\big(f_{\text{CLIP}}(I), \mathbf{f}_{F(I)}\big)$.

Theoretical implication: Bad views tend to be scored poorly (ambiguous), and therefore are much more likely to be selected as I_{amb}. Thus the probability of encountering a good–bad pair increases automatically.

Algorithm 5 LLM-Polygon: Spatially Grounded Multi-View Label Refinement

Require: Detection images $\mathcal{I}$, camera angles $\mathcal{P}$, polygon faces $\mathcal{F}$, uncertainty weight λ

Initialize hypothesis cache $\mathcal{C} \leftarrow \varnothing$
$t \leftarrow 0$
while $|\mathcal{I}| \geq 2$ and $t < T_{\max}$ **do**
  $t \leftarrow t + 1$
  Spatial assignment of detections:
  for each face $F \in \mathcal{F}$ **do**
    $\mathcal{I}_F \leftarrow$ detections whose camera rays intersect F
  end for
  Compute per-face features and uncertainties:
  for each face $F \in \mathcal{F}$ **do**
    if $\mathcal{I}_F$ is non-empty **then**: $\mathbf{f}_F \leftarrow \frac{1}{|\mathcal{I}_F|} \sum_{I \in \mathcal{I}_F} f_{\text{CLIP}}(I)$
    else: $\mathbf{f}_F \leftarrow \lambda \cdot \mathbf{u}$ ▷ $\mathbf{u}$: calibrated uncertainty vector
    end if
  end for
  Select informative views: $\forall \mathcal{I}$: compute $s_I = \cos(f_{\text{CLIP}}(I), \mathbf{g})$
  $I_{\text{rep}} \leftarrow \arg\max_{I \in \mathcal{I}} s_I$ ▷ closest to mean, most representative
  $I_{\text{amb}} \leftarrow \arg\min_{I \in \mathcal{I}} s_I$ ▷ farthest from mean, most ambiguous
  $\mathcal{I}_{\text{sample}} \leftarrow \{I_{\text{rep}}, I_{\text{amb}}\}$
  Compute spatially grounded global confidence:
  $c_{\text{global}} \leftarrow \frac{1}{|\{F : \mathcal{I}_F \neq \varnothing\}|} \sum_{F : \mathcal{I}_F \neq \varnothing} \cos(h, \mathbf{f}_F)$
  Determine which image harms consistency more:
  if $s_{amb} < s_{rep}$ **then**: Remove I_{amb} from $\mathcal{I}$
  else: Remove I_{rep} from $\mathcal{I}$
  end if
  Cache hypothesis: Store $(\hat{y}, h, c_{\text{global}})$ in $\mathcal{C}$
  Select next viewpoint via spatial uncertainty:
  For each face F, compute Δ_F = max confidence difference with adjacent faces
  $P_{\text{next}} \leftarrow$ viewpoint observing face with largest Δ_F, prioritizing unseen faces
end while
Recover most confident hypothesis: $(y^*, h^*, c^*) \leftarrow \arg\max_c (\hat{y}, h, c) \in \mathcal{C}$
return (y^*, P_{next})

(C) Spatially grounded viewpoint selection. The next viewpoint P_{next} is chosen from faces with large confidence differences Δ_F, prioritizing unseen faces.

Theoretical implication: Exploration becomes *systematic and directed*, guaranteeing eventual exposure of faces where inconsistencies (bad views) are likely to reside.

LLM-Polygon is not simply LLM-Sampling with additional notation. Its geometric structure fundamentally changes the sampling distribution, leading to provably stronger constants in the convergence theorem. We now formalize these improvements.

Theorem 3 (Exponential Decay of Bad Views under LLM-Polygon) *Let G_t and B_t denote the good and bad views at iteration t, and $b_t = |B_t|$. Assume:*

1. ***Semantic dominance:*** *For every good–bad pair $\{I_g, I_b\}$ selected as $I_{\text{rep},t}, I_{\text{amb},t}$,*

$$\Pr[\cos(h_{t+1}, f(I_g)) > \cos(h_{t+1}, f(I_b))] \geq \eta > \tfrac{1}{2}.$$

2. ***Geometric exposure of bad views:*** *For every iteration with $b_t > 0$,*

$$\Pr[\{I_{\text{rep},t}, I_{\text{amb},t}\} \cap B_t \neq \varnothing] \;\geq\; \delta_{\text{poly}} > 0.$$

That is, with probability at least δ_{poly}, the deterministically chosen representative and ambiguous views include at least one bad view. This probability is enforced by polygon-based partitioning, per-face feature averaging, and spatial uncertainty prioritization.

Then the probability of retaining any bad views after T iterations satisfies $\Pr[b_T > 0] \leq c_1 e^{-c_2 T}$, for constants $c_1, c_2 > 0$ depending only on η and δ_{poly}.

In particular, the algorithm eliminates all bad views after $O(\log(1/\varepsilon))$ iterations with probability at least $1-\varepsilon$.

Remark. The proof follows the same Markov-chain drift argument as for the non-polygon algorithm, with the key constant β replaced by $\beta_{\text{poly}} \geq \eta\,\delta_{\text{poly}}$, where δ_{poly} is the probability that the deterministically selected pair $(I_{\text{rep},t}, I_{\text{amb},t})$ forms a good–bad pair. Since polygon-based spatial partitioning and per-face averaging ensure $\delta_{\text{poly}} > \delta_{\text{sample}}$, the drift toward eliminating bad views is strictly larger, yielding a strictly larger exponential rate c_2 in the bound

$$\Pr[b_T > 0] \leq c_1 e^{-c_2 T}.$$

6 Experiments

We evaluate our proposed method, LADR, against several baseline algorithms in both single-object and multi-object settings. The experiments are designed to assess each method's ability to infer object semantic labels accurately in multi-view scenarios. On single-object scenes, we demonstrate that reasoning with a large language model is crucial for 3D consistency and that active view selection greatly improves sample efficiency and stability. We also evaluate all methods on multi-object scenes, a realistic setting for robot exploration. Here, we isolate the impact of our label generation mechanism by using off-the-shelf exploration policies rather than proposing next-best views. This setup highlights that our representation still improves multi-object labeling as an offline refinement process. Full details of all hyperparameters used are provided in Appendix B.1, and extended results are presented in B.4 and B.5.

6.1 Baselines

We compare methods that rely solely on YOLO detections, CLIP embeddings, or LLM reasoning, with LADR, which leverages multi-view aggregation, spatial grounding, and confidence-based label selection.

'YOLO': uses the most common YOLO label as the final label. This is the aggregation policy in ConceptFusion (Jatavallabhula et al., 2023).

'CLIP': takes the average of the CLIP embeddings of all images and compares it via cosine similarity to an extensive list of CLIP-embedded labels of the RAM class list (Zhang et al., 2023).

'LLM-Label': The LLM reasons over the set of YOLO labels, their frequencies, and possible semantic relationships to infer the most plausible label. No visual data is used, only text.

'LLM-Tiled': creates a single image with all input images tiled. This layout is then analyzed by a large vision model to produce the final label. (We provide example in Appendix B.6).

'LLM-Angle': creates a single composite image with all input images around a circle capturing their relative positions to the object. This panoramic-style layout is then analyzed by a large vision model to produce the final label. Unlike the other baselines, the LLM provides the next best view to take an image from. (We provide example in Appendix B.6).

LADR implementation: Here, LADR refers to our three algorithms: LLM-Random, LLM-Sampling, and LLM-Polygon. Apart from these, only 'LLM-Angle' explicitly proposes the next best view; for all other baselines, random view sampling is used when not otherwise specified.

6.2 Dataset

We evaluate our methods on a single-object dataset, a subset of the *OmniObjects3D* (Wu et al., 2023) dataset of annotated 3D object models. These objects are rendered in NVIDIA Isaac Sim under controlled conditions to generate multi-view image sequences. We focus on five object classes: backpack, cup, cabinet, sofa, and suitcase. For each class, we include five distinct instances, several of which are deliberately misleading in appearance (e.g., a mug shaped like a cartoon character) to test the robustness of semantic labeling methods.

We also constructed a multi-object dataset in the same simulation environment. These scenes contain multiple objects arranged in varied environments, including SimpleRoom, Commercial, Industrial, Residential, and Vegetation, providing more complex scenarios with occlusions.

Each object in the datasets is annotated with both its class name and a concise descriptive phrase, for example a chair labeled as *chair* with the description *wooden dining chair with a cushioned seat*. We provide examples for both datasets in Appendix B.3.

6.3 Evaluation Metrics

To assess performance, the predicted labels are compared against ground-truth object class names and longer, descriptive phrases (e.g., "yellow cartoon character-shaped mug"). Since LADR is an open-vocabulary setting, direct comparison with ground-truth labels is not sufficient: the LLM may propose synonyms of the annotated class, which should be accepted. Empirically, we found that the CLIP model used for image–text similarity is overly sensitive to lexical variation (e.g., number of words), leading to unreliable synonym matching. Instead, we employ a Sentence Transformer (Reimers & Gurevych, 2019) model to evaluate label equivalence. The final similarity score is defined as the maximum of the similarity to the class name and the similarity to the description, capturing both category-level and instance-level alignment. To evaluate success rates rather than raw similarities, we adopt the similarity value 0.5 as the threshold for label correctness (based on preliminary experiments; see Appendix B.2), while also considering thresholds of 0.3, 0.7, and 0.9.

To evaluate detections in the multi-object setting, we establish one-to-one matches between ground-truth objects and predicted detections from the global map. Matching is based on a semantic-spatial similarity score, defined as a weighted sum of label similarity and spatial overlap between ground-truth and predicted bounding boxes. Once matches are established, evaluation metrics follow the same procedure as in the single-object setting, ensuring comparability.

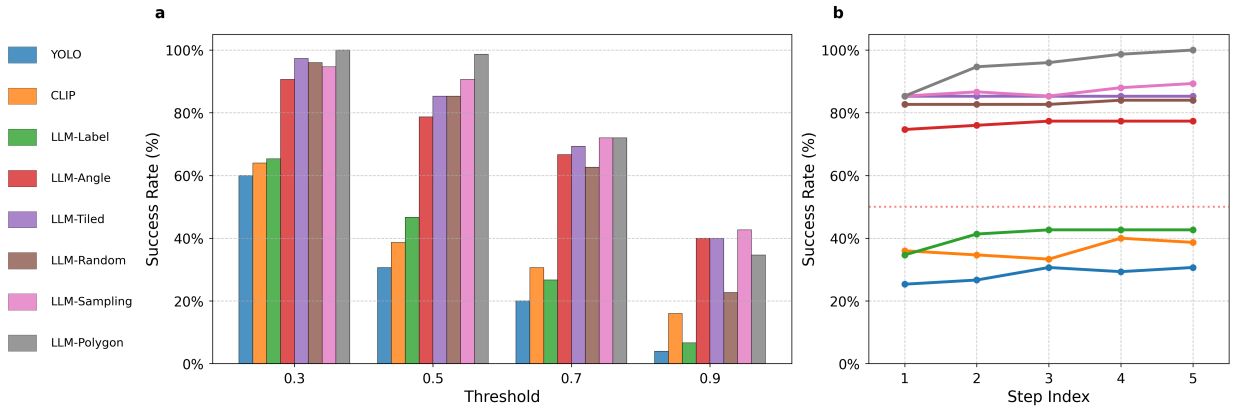

Figure 3: **Single-Object Experiment Results** (a) Averaged success rates across different success thresholds for each algorithm. (b) Evolution of success rates over data collection steps for each algorithm, using 0.5 as the threshold.

6.4 Summary of Findings

We provide our results for the single- and multi-object cases in Figures 3 and 4, respectively. We provide detailed results, including per-object examples for each setting in Appendices B.4 and B.5. Figure 3a shows

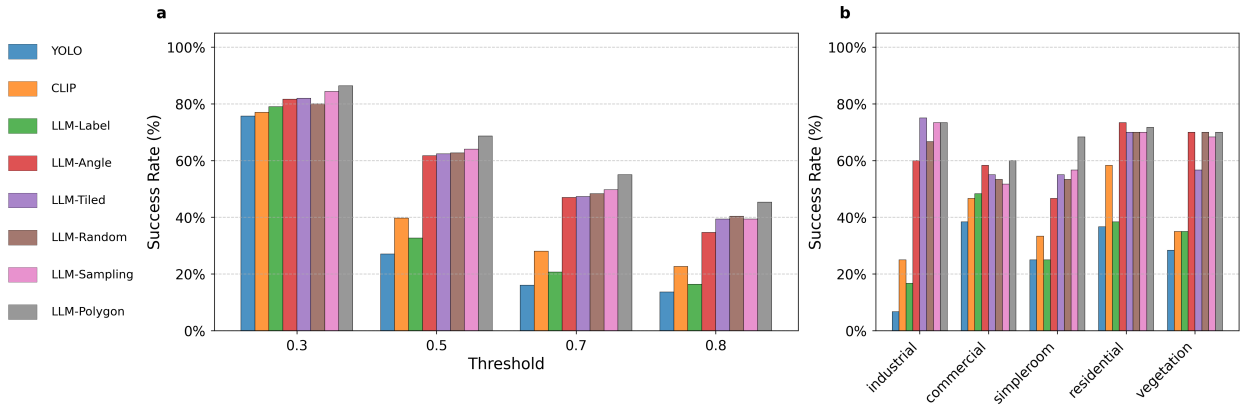

Figure 4: **Multi-Object Experiment Results** (a) Averaged success rates across different success thresholds. (b) Averaged success rates across scenes, using 0.5 as the threshold.

the averaged success rates based on different success thresholds, and Figure 3b shows how success rates evolve over the data collection steps with 0.5 as the threshold.

Similar trends are observed for single- and multi-object cases. The first observation is that they show over 40% improvements, respectively, compared to ubiquitous fusion methods using YOLO, and CLIP. **YOLO** and **LLM-Label** rely solely on YOLOE predictions, resulting in consistently low success rates. This is likely due to their lack of multi-view image-based reasoning. Notably, LLM reasoning alone offers little improvement over simply taking the most frequent YOLO label. **CLIP** performs comparably to YOLO, but struggles with the vast label set and ambiguity introduced by averaging embedded views, often leading to confused predictions. **LLM-Tiled** achieves higher success rates by leveraging all views simultaneously. However, its accuracy lags behind LADRs, suggesting that the tiled representation loses fine-grained detail or introduces structural incoherence that limits reasoning. **LLM-Angle** adds structural consistency by ordering views in a layout, yet provides no improvement over LLM-Tiled. This indicates that the performance gap is more likely due to loss of visual detail. **LLM-Random** and **LLM-Sampling** analyze images in greater detail, leading to stronger descriptive accuracy. However, LLM-Random often declares detections prematurely, and LLM-Sampling cannot fully mitigate instability despite its confidence-based pruning. Finally, **LLM-Polygon** outperforms all, with near-perfect success at a 0.5 threshold. By combining detailed reasoning with active exploration and consistency across unseen sides, it avoids the pitfalls of LLM-Only and LLM-Sampling. Figure 3 /b shows how active exploration of unseen sides leads to success rate improvement. LADR's combination of uncertainty sampling, confidence computation, and spatial grounding is key to outperform approaches that provide multiple images to an LLM, as in LLM-Tile and LLM-Angle.

7 Conclusion

Our contributions in this work center on a scalable framework for iterative sampling with LLM-guided active refinement and exploration in open-vocabulary 3D object detection. By integrating the generative reasoning of LLMs with the quantitative similarity assessment of contrastive VLMs, our approach substantially improves label consistency, establishing a foundation for future research in robust and efficient 3D perception. The method also serves as a drop-in extension to existing object detection pipelines, allowing zero-shot re-evaluation of detections.

Despite these advantages, the methods presented require multiple inner-loop queries, which increases computational cost. This limitation could be mitigated through batch sampling strategies, or pruning multiple detections simultaneously, as well as by employing more efficient vision-language models, e.g., FastVLM (Vasu et al., 2025), to enable inference on resource-constrained hardware.

References

Vincent Cartillier, Zhile Ren, Neha Jain, Stefan Lee, Irfan Essa, and Dhruv Batra. Semantic mapnet: Building allocentric semantic maps and representations from egocentric views, 2021. URL https://arxiv.org/abs/2010.01191.

Devendra Singh Chaplot, Murtaza Dalal, Saurabh Gupta, Jitendra Malik, and Ruslan Salakhutdinov. Seal: Self-supervised embodied active learning using exploration and 3d consistency. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. —, 2021. arXiv:2112.01001.

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection, 2024. URL https://arxiv.org/abs/2401.17270.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2829. IEEE, June 2023. doi: 10.1109/cvpr52729.2023.00276. URL http://dx.doi.org/10.1109/CVPR52729.2023.00276.

Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. In *Proceedings of Robotics: Science and Systems (RSS)*, February 2023. arXiv:2302.07241.

Christina Kassab, Matías Mattamala, Sacha Morin, Martin Büchner, Abhinav Valada, Liam Paull, and Maurice Fallon. The bare necessities: Designing simple, effective open-vocabulary scene graphs. *arXiv preprint*, abs/2412.01539, Dec 2024. arXiv:2412.01539.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. URL https://arxiv.org/abs/2304.02643.

OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–824, June 2023. doi: 10.1109/CVPR52729.2023.00085.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos, 2024. URL https://arxiv.org/abs/2408.00714.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 779–788, 2016.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild, 2022. URL https://arxiv.org/abs/2204.07761.

Gianluca Scarpellini, Stefano Rosa, Pietro Morerio, Lorenzo Natale, and Alessio Del Bue. Look around and learn: Self-training object detection by exploration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. arXiv:2302.03566.

Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokul Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, and Hadi Pouransari. Fastvlm: Efficient vision encoding for vision language models, 2025. URL https://arxiv.org/abs/2412.13303.

Ao Wang, Lihao Liu, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yoloe: Real-time seeing anything, 2025. URL https://arxiv.org/abs/2503.07465.

Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, Dahua Lin, and Ziwei Liu. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation, 2023. URL https://arxiv.org/abs/2301.07525.

Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, Yandong Guo, and Lei Zhang. Recognize anything: A strong image tagging model, 2023. URL https://arxiv.org/abs/2306.03514.