ZeroKey: Point-Level Reasoning and Zero-Shot 3D Keypoint Detection from Large Language Models

Bingchen Gong[1] **Diego Gomez**[1] **Abdullah Hamdi**[2] **Abdelrahman Eldesokey**[3]
Ahmed Abdelreheem[3] **Peter Wonka**[3] **Maks Ovsjanikov**[1]

[1]École Polytechnique [2]Visual Geometry Group, University of Oxford
[3]King Abdullah University of Science and Technology (KAUST)

gongbingchen@gmail.com

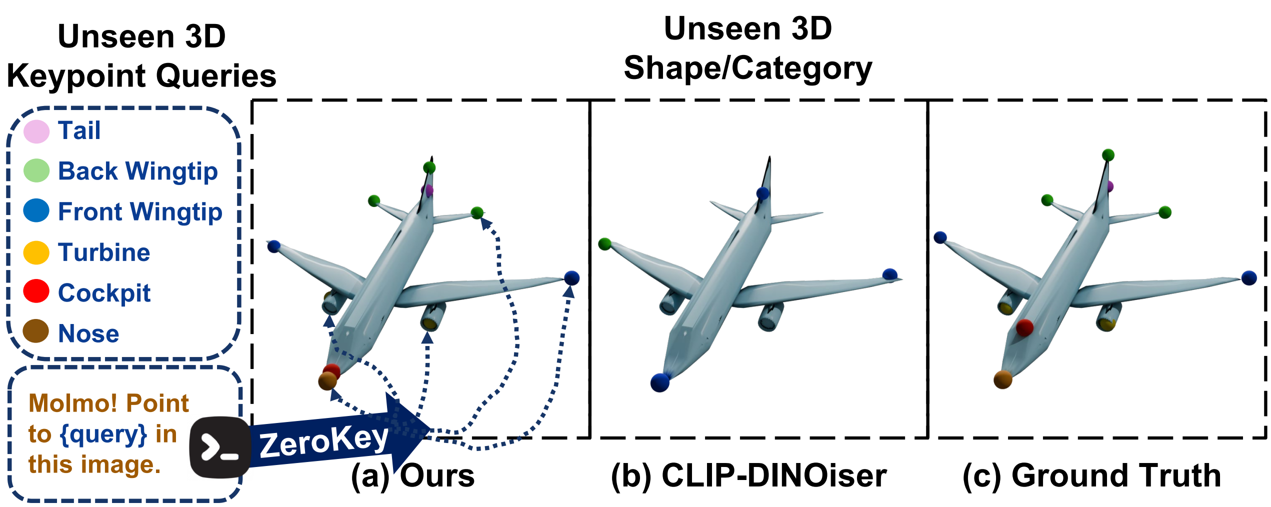


Figure 1. **Zero-shot 3D Keypoint Detection.** Without any ground truth labels or supervised training, our method leverages the point-level reasoning embedded within MLLMs to extract and name salient keypoints on 3D models. The figure illustrates how our approach achieves competitive performance compared to CLIP-DINOiser [50] baselines, highlighting the potential of integrating language models with vision tasks for enhanced 3D shape understanding.

Abstract

We propose a novel zero-shot approach for keypoint detection on 3D shapes. Point-level reasoning on visual data is challenging as it requires precise localization capability, posing problems even for powerful models like DINO or CLIP. Traditional methods for 3D keypoint detection rely heavily on annotated 3D datasets and extensive supervised training, limiting their scalability and applicability to new categories or domains. In contrast, our method utilizes the rich knowledge embedded within Multi-Modal Large Language Models (MLLMs). Specifically, we demonstrate, for the first time, that pixel-level annotations used to train recent MLLMs can be exploited for both extracting and naming salient keypoints on 3D models without any ground truth labels or supervision. Experimental evaluations demonstrate that our approach achieves competitive performance on standard benchmarks compared to supervised methods, despite not requiring any 3D keypoint annotations during training. Our results highlight the potential of integrating language models for localized 3D shape understanding. This work opens new avenues for cross-modal learning and underscores the effectiveness of MLLMs in contributing to 3D computer vision challenges.

1. Introduction

Multimodal Large Language Models (MLLMs) have been shown to seamlessly integrate visual and text representations with great success [3, 27–29, 34, 59]. Models like GPT-4 with vision capabilities, OFA [48], and Flamingo [3] have demonstrated the ability to process visual and textual data, leading to advancements in vision-language understanding. Furthermore, related efforts have been made to endow MLLMs with 3D or spatial reasoning [18, 51, 58]. When trained at scale, these MLLMs enable tackling a wide array of complex tasks, ranging from comprehensive image or 3D shape description to answering detailed questions about visual content.

Unfortunately, despite this tremendous progress, existing MLLMs still struggle with tasks that require precise points or pixel-level reasoning, such as localization, counting, or salient keypoint detection [8, 37, 56]. This type of reasoning focuses on understanding and interpreting visual input at a fine-grained level using text. In general, we can observe an increased level of difficulty when going from complete objects to object parts and, finally, to specific *points* or small regions. Even advanced models like GPT-4o [34] and Claude Sonnet 3.5 [4], which have demonstrated impressive capabilities in various computer vision tasks, still struggle with point-level understanding.

This challenge arises because point-level reasoning requires models to accurately identify and analyze local details, such as landmarks on a face, joints in a human pose, or intricate components in a 3D shape. However, the architectures, training methods, and dataset annotations of most existing approaches are primarily designed to capture global visual properties and potentially link them to textual descriptions. Since text captions are most often crawled from the Internet, and thus rarely contain precise point or region-level annotations, point-level reasoning capabilities of most MLLMs remain very limited.

Addressing this problem necessitates a shift in how models are designed and trained. One potential solution is to use more powerful visual encoders trained on real-world data encompassing a variety of global and local features. Interestingly, scaling alone to more crawled data or model complexity does not seem to alleviate this problem [35]. In contrast, very recently, a family of MLLMs has been introduced that was trained with specially-designed auxiliary training data and tasks **aimed at pixel-level image annotations** *alongside* global image captions [12]. When trained with this additional data source, the resulting model (dubbed Molmo) exhibits impressive localized reasoning abilities (counting, localization, etc.) surpassing other MLLMs.

Inspired by these recent developments, we propose investigating MLLMs endowed with point-level reasoning in the context of 3D shape understanding and specifically for zero-shot keypoint detection. Given a 3D model from an arbitrary category, our main task is to localize and name salient keypoints on this model. Traditional methods for 3D keypoint detection heavily rely on annotated 3D datasets and extensive supervised training [5, 49, 55]. This severely limits their applicability to specific data-rich object categories. At the same time, recent general-purpose 3D LLMs are typically trained with explicit objectives that align 3D data representations with vision-language models [18], enabling global tasks like classification. However, since they are based on standard vision-language models with limited localization capabilities, none of these methods can accurately perform 3D keypoint localization based on a language prompt, see the supplementary material for more details.

Even in the presence of a powerful MLLM, we note that localization or keypoint detection from images to 3D environments presents a significant challenge. Since both detections and point-level captions tend to be noisy, developing a robust approach requires reliable multi-view aggregation, backprojection, and filtering mechanisms. In this paper, we propose a comprehensive zero-shot 3D keypoint detection method that exploits the point-level reasoning capabilities of recent MLLMs [12] while integrating 3D consistency and being completely category agnostic. To the best of our knowledge, ours is the first robust method for zero-shot 3D keypoint detection.

Furthermore, we analyze the 3D understanding encoded in Molmo through our method by leveraging Schelling Points and evaluating the describability of keypoints. Through this study, we characterize the strengths and limitations of the 3D awareness imparted to models through training with pixel-level annotations. Our evaluations demonstrate that, despite not having access to ground truth data, our method can predict points similar to those identified by human annotators by selecting appropriate text prompts. These two tasks could serve as valuable evaluations for the point-level reasoning capability of multi-layered language models (MLLMs).

Additionally, we developed two text-image baselines for detecting 3D keypoints from text prompts and conducted comparisons with these baselines.

To summarize, our main contributions are as follows:

- Identify the challenge of zero-shot point-level reasoning in 3D computer vision.
- Propose the first zero-shot 3D keypoint detection method that operates without 3D annotations or training and applies to arbitrary shape categories, establishing a baseline for zero-shot 3D keypoint detection in this new area.
- Evaluate several MLLMs in our context, highlighting the importance of auxiliary point-level training to solve the task at hand.

2. Related Work

2.1. Multimodal Large Language Models

The remarkable success of vision-language models like CLIP [36] and large language models like GPT-3 [7] has motivated the integration of multimodal capabilities into large language models for visual understanding. Multimodal Large Language Models (MLLMs) have been extended to process images alongside text, achieving remarkable success in vision-language tasks [3, 27–29, 34, 59].

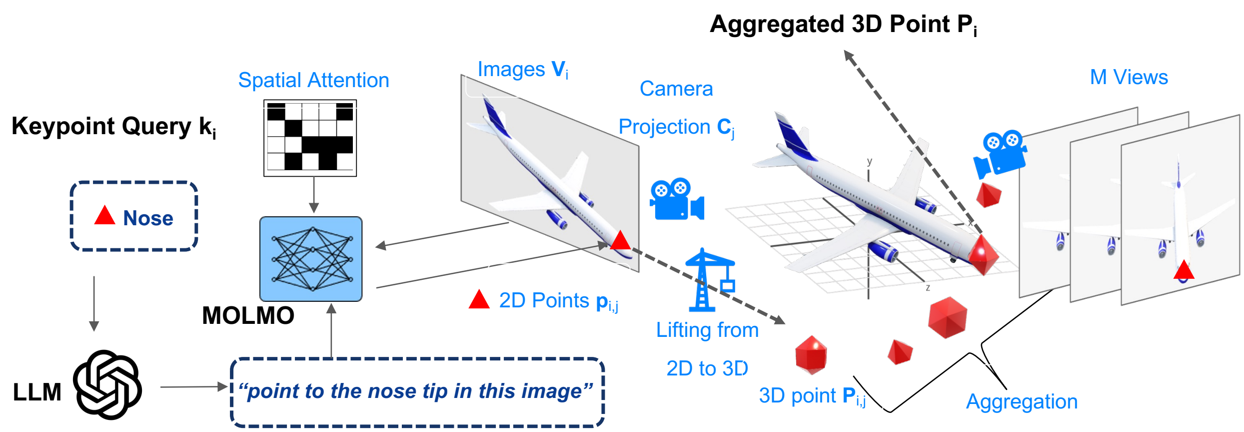


Figure 2. **ZeroKey Pipeline.** Our proposed ZeroKey employs MLLM Molmo for zero-shot keypoint detection on 3D objects by 1) rendering multiple views for a given shape, 2) leveraging MLLM reasoning in each view using point-specific prompts, and 3) aggregating the results through clustering, eliminating the need for annotated training data for 3D keypoint detection.

Notable models such as GPT-4 with vision capabilities, MiniGPT-4 [59], LLaVA [28, 29], Flamingo [3], and BLIP [27] have demonstrated advanced abilities in integrating visual and textual information, enabling tasks ranging from detailed image captioning to complex visual reasoning. These models leverage large-scale pre-training on extensive image-text datasets to capture both global semantics and fine-grained details in 2D images. Despite their success with 2D data, extending these capabilities to 3D data remains a significant challenge due to the elevated cost and complexity of collecting large amounts of high-quality 3D data. Consequently, there is no dominant MLLM capable of processing 3D data with natural language understanding. To address this gap, recent efforts have focused on aligning models trained on small amounts of 3D data with large vision-language models. For instance, ULIP [52] seeks to align 3D models with vision-language representations to enhance cross-modal understanding. Similarly, Hong et al. [18] propose training a 3D Large Language Model by leveraging a pre-trained vision-language model, aiming to bridge the modality gap and facilitate better 3D comprehension through multimodal learning. While these "3D LLMs" are promising, the granularity we are targeting in work at the point-level 3D reasoning is still beyond their capabilities [18, 52]. See the supplementary material for more details.

2.2. Keypoint Detection

Keypoint detection has been extensively studied in 3D due to its significance in shape representation, matching, and abstraction applications [9, 13, 21, 41, 44, 47, 53]. However, the definition of a keypoint has remained task-specific and subjective. Even human annotations of keypoints often differ, as highlighted in [10, 54]. Recent work [17] has shown that image diffusion models can develop an unsupervised understanding of keypoints as salient features. Additionally, it was demonstrated in [42] that the CLIP foundation model [36] can reasonably interpret keypoints with appropriate visual prompts. This paper investigates the understanding of keypoints as foundational elements of objects and shapes within recent multimodal large language models (MLLMs).

2.3. Lifting from 2D to 3D

A number of studies have explored multi-view approaches to enhance 3D data analysis by leveraging 2D representations [11, 14, 16, 22, 23, 25, 30, 32, 33, 46]. A key challenge in this area is combining features from different viewpoints to represent local 3D points or voxels while preserving essential geometric details. Some methods address this by averaging features across views [23, 25], propagating labels between views [46], or learning from reconstructed neighboring points [22]. Others organize points within a unified grid structure [31] or merge multi-view features with 3D voxel information [11, 19]. VointNet [15] operates within the Voint cloud framework, retaining the original point cloud's compactness and descriptive 3D properties. It employs learned aggregation of multi-view features applied independently to each point. Recently, there's a growing trend of utilizing foundational models with zero-shot understanding capabilities—such as the Segment Anything Model [24]—to achieve zero-shot 3D comprehension by lifting 2D predictions into 3D space [1, 2, 20, 40]. Our work differs from the Zero-shot 3D segmentation in that our final goal is a fine-grained *single* 3D point representing an unseen text query on an unseen new 3D object. As we show below, existing vision-language models struggle with this task, thus necessitating a novel adapted solution.

3. Motivation

Localization and naming of points in an image or a 3D shape is an extremely challenging problem. Solving such a problem requires a localized understanding of the visual data and tight integration between text. Vision-Language Models (VLMs) are thus a strong candidate to tackle this problem. However, as mentioned above, most such models are trained using global or object-level semantics which are insufficient for point localization tasks. Furthermore, the recent MLLMs that incorporate 3D data [18, 51, 58] are typically trained with explicit alignment against pre-trained traditional vision-language models, and thus inherit their limitations in point-level reasoning. As a result, such models are unable to solve the zero-shot keypoint detection problem, which is the main focus of our work.

Existing methods for the 3D keypoint detection problem typically formulate the problem as either a supervised learning task, by exploiting the ground truth annotations, e.g., in the KeypointNet dataset [54]. Alternatively, several few-shot approaches have been developed, e.g., [49], which are based on *transferring* known keypoints to new instances.

Rather than training an entirely new 3D model, our goal is to exploit the improved point-level reasoning capabilities of very recent VLMs and demonstrate their utility in 3D shape understanding tasks. We base our approach on the recent open-source MOLMO model [12], which incorporated a specially designed pixel-level annotation task into its training. Thanks to a well-chosen design and this additional curated dataset the model has been shown to perform significantly better than even the largest closed-source models in localized image understanding.

Our key motivation is to evaluate the effectiveness of such a model in the context of *3D shape* understanding and specifically for zero-shot keypoint detection. We first introduce the problem of zero-shot keypoint detection and naming. Then, we develop a pipeline based on prompting the model and multi-view aggregation. We show that the resulting approach leads to unprecedented localized 3D shape understanding, opening the door to a range of applications. In this paper, we present the very first baseline for solving the zero-shot 3D keypoint detection problem. We leverage the powerful 2D spatially-aware VLM, Molmo [12], to generate precise 2D keypoint predictions from several points of view. These points are back-projected into a 3D shape, and after aggregation, we obtain a 3D keypoint prediction.

4. Method

Problem Setting and Overview. We consider the Zero-Shot Keypoint Detection problem for 3D shapes. Specifically, given a 3D shape, we aim to automatically generate a set of salient points corresponding to the shape's semantic parts. Our solution comprises three main components: first, we prompt a MLLM with the shape, asking the model to generate a list of names for possible candidate keypoints. Then, for each candidate, we ask the model to detect the precise coordinates of the point in a given image. Finally, we back-project those detected points into 3D and aggregate them to get the location of 3D keypoints in a given shape.

4.1. Generating Text Candidates for Salient Points

To detect points using a MLLM, we first need to assign names to these points. The definition of keypoints can vary depending on the application. However, these points are typically common to a given shape, salient, and carry significant semantic meaning. This information can be retrieved from MLLMs by prompting the model with the shape class in text form or by using an image of the shape when the MLLM supports multimodal input. Given the shape class label c, we prompt the MLLM to generate a list of candidate keypoint names: $\mathcal{K} = \{k_1, k_2, \ldots, k_N\}$, where N is the number of keypoints generated. For example, for the class "airplane," the MLLM might generate keypoint names such as "nose," "wing tip," and "tail." In our experiments, we utilize GPT-4o [34] as our MLLM backbone for this step. We prompt it with a front-facing rendering of the 3D model along with the request text: *"List possible salient key points (in text)."* The GPT-4o model is instructed to provide a candidate list of possible names for keypoints in JSON format.

4.2. Prompting Molmo to Detect 2D Keypoints

To detect the precise 2D coordinates of each candidate keypoint, we utilize Molmo [12], a state-of-the-art MLLM capable of localizing points in images based on natural language prompts. Given a set of candidate keypoint names $\mathcal{K} = \{k_1, k_2, \ldots, k_N\}$ and a set of images (views) of the 3D shape $\mathcal{V} = \{\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_M\}$, we prompt Molmo to detect the location of each keypoint in each image. We define the detection function as:

$$\mathbf{p}_{i,j} = \text{Molmo}(\mathbf{V}_j, k_i), i = 1, \ldots, N; j = 1, \ldots, M, \quad (1)$$

where $\mathbf{p}_{i,j} \in \mathbb{R}^2$ are the 2D coordinates of keypoint k_i in image $\mathbf{V}_j$ as predicted by Molmo. The prompt to Molmo consists of the image $\mathbf{V}_j$ and the instruction to localize the keypoint k_i. For example:

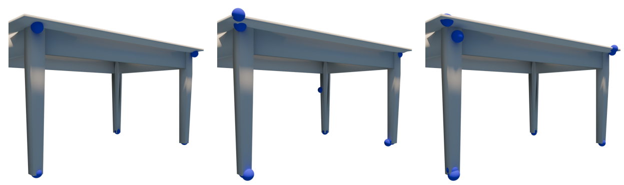
"Point to the left wing tip in this image."

This leverages Molmo's capability to understand natural language instructions and perform point-level localization.

4.3. Zero-Shot 3D Keypoint Detection

After obtaining the 2D keypoint detections from Molmo, we reconstruct the 3D coordinates of each keypoint in shape by back-projecting the 2D points into 3D space and aggregating the results from multiple views. Fig. 3 shows aggregated points from different numbers of views.

Back-Projection of 2D Keypoints. Given the camera projection matrices $\{\mathbf{C}_j\}_{j=1}^M$ corresponding to each image $\mathbf{V}_j$,



(a) 6 Views (b) 26 Views (c) 46 Views

Figure 3. The number of rendered views versus the detected keypoints after aggregation. This figure shows how varying the number of rendered views affects the total number of keypoints detected by ZeroKey. The prompt here is *"corner of the table"*.

we can back-project the 2D points $\mathbf{p}_{i,j}$ into 3D space. Assuming a pinhole camera model, each camera projection matrix $\mathbf{C}_j$ maps 3D points $\mathbf{P} \in \mathbb{R}^3$ to 2D points $\mathbf{p} \in \mathbb{R}^2$:

$$\mathbf{p}_{i,j} \sim \mathbf{C}_j \mathbf{P}_{i,j}, \quad i = 1, \dots, N; \quad j = 1, \dots, M, \qquad (2)$$

where $\sim$ denotes equality up to scale. To back-project $\mathbf{p}_{i,j}$ into 3D space, we compute the ray $\mathbf{r}_{i,j}(t)$ corresponding to $\mathbf{p}_{i,j}$ as $\mathbf{r}_{i,j}(t) = \mathbf{C}_j^{-1}(\mathbf{p}_{i,j}, t)$, where t represents the depth along the ray. We intersect the ray with the 3D shape S to find the 3D point $\mathbf{P}_{i,j} = \text{Intersect}(\mathbf{r}_{i,j}, S)$. During the rasterization process, we cache the depth t for each pixel and utilize that information for back-projection, eliminating the need for expensive intersection calculations.

Soft Voting Weights. To handle sharp angular intersections and stabilize the mapping process, we back-project a $h \times h$ patch $\mathbb{S}_{i,j}$ in the region centered at $\mathbf{p}_{i,j}$. We exclude all pixels that do not intersect with the mesh, then compute the mean of the back-projected points within the patch to obtain a stable back-projection of $\mathbf{P}_{i,j}$. We also assign a soft-voting weight $\mathbf{W}_{i,j}$ to each $\mathbf{P}_{i,j}$ derived from the Gaussian kernel centered at $p_{i,j}$:

$$\mathbf{W}_{i,j} = \sum_{\mathbf{p}_{m,n} \in \mathbf{N}_{i,j}} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\left\|\mathbf{p}_{i,j} - \mathbf{p}_{m,n}\right\|^2}{2\,\sigma^2}\right), \qquad (3)$$

where $\mathbf{N}_{i,j} = \{\mathbf{p}_{m,n} | \mathbf{p}_{i,j} \in \mathbb{S}_{m,n}\}$. The point $\mathbf{P}_{i,j}$ gains higher weights when its corresponding pixel $\mathbf{p}_{i,j}$ in the patch $\mathbb{S}_{m,n}$ is closer to the center pixel $\mathbf{p}_{m,n}$. The σ is set to $\frac{h}{3}$ in our implementation.

Aggregation of 3D Keypoints. For each of the N keypoints k_i, we collect the set of back-projected 3D points from all views: $\mathcal{P}_i = \{\mathbf{P}_{i,j} \mid j = 1, \dots, M\}$. To obtain the final 3D keypoint location $\hat{\mathbf{P}}_i$, we aggregate the set $\mathcal{P}_i$. A straightforward method is to compute the weighted mean of the points $\hat{\mathbf{P}}_i = \frac{\sum_{j=1}^{M} \mathbf{W}_{i,j}\mathbf{P}_{i,j}}{\sum_{j=1}^{M} \mathbf{W}_{ij}}$. However, naive summarization often produces unsatisfactory results. This occurs because back-projection may become unstable when the angle between the ray and the mesh intersection is sharp. In such cases, the ray may intersect different parts of the object, leading to inconsistencies. Molmo [12] can also generate multiple keypoints for one prompt, resulting in ambiguity and inconsistencies in the summarized coordinates. Patchify back projection can help to address the issue of sharp angular intersections by providing a weight for each back-projected point. One can also extract the confidence weighting from Molmo's feature map and apply it to the predictions. Given that the predictions from Molmo can produce multiple points for each prompt and may seem noisy from different perspectives, we treat Molmo as a black box in our approach. We aggregate all the points into a single point cloud, which contains an unknown number of key points. Subsequently, we apply hierarchical density-based spatial clustering to this aggregated point cloud.

Filtering and Refinement. Specifically, we apply HDBSCAN to the set of predicted points $\mathcal{P}_i$ obtained from multiple views to identify dense clusters that represent consistent predictions across views. HDBSCAN operates by defining the mutual reachability distance d_{m} between two points $\mathbf{P}_{i,j}$ and $\mathbf{P}_{i,l}$ as:

$$d_{\text{m}} = \max\left(\text{core}_k(\mathbf{P}_{i,j}),\ \text{core}_k(\mathbf{P}_{i,l}),\ d(\mathbf{P}_{i,j}, \mathbf{P}_{i,l})\right), \qquad (4)$$

where $\text{core}_k(\mathbf{P}_{i,j})$ is the core distance of point $\mathbf{P}_{i,j}$, defined as the distance to its k-th nearest neighbor, and $d(\mathbf{P}_{i,j}, \mathbf{P}_{i,l})$ is the Euclidean distance between points $\mathbf{P}_{i,j}$ and $\mathbf{P}_{i,l}$. We apply each point $\mathbf{P}_{i,j}$ corresponding weights $\mathbf{W}_{i,j}$ obtained from Eq.3. Throughout our experiments, we consistently set minPts $k = 10$. By employing the mutual reachability distance with this fixed minPts parameter, we effectively filter out noise and robustly identify the most stable clusters among the predicted points.

5. Practical 3D Understanding with MLLMs

Gauging to what extent MLLMs are able to reason about 3D structures remains an underexplored question with significant implications and applications. Molmo [12] has demonstrated strong 2D spatial reasoning abilities. We find that this reasoning generalizes to some degree to 3D. In this section, we propose to probe Molmo's 3D reasoning capabilities by leveraging Schelling points and describability. Through this, we aim to characterize the strengths and limitations of its 3D reasoning, specifically the strengths and limitations of the 3D reasoning induced by training with pixel-level annotations.

5.1. Schelling Points on 3D Surface

Schelling points, as introduced by Schelling [39], are specific focal points that people are likely to choose independently due to their prominence. In visual perception, these keypoints stand out due to distinct features. We hypothesize that the salient points people select often have clear, widely understood names in language. Therefore, an effective MLLM should be able to better predict such salient points by using its knowledge of both language and vision.

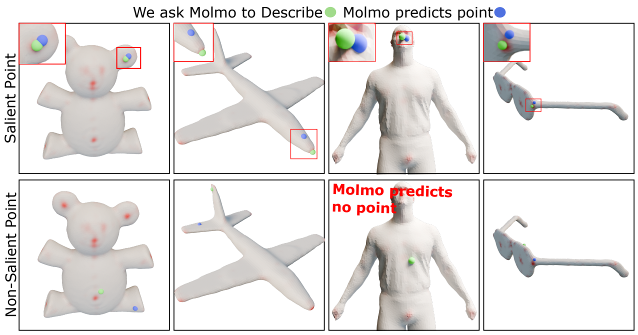


Figure 4. We ask Molmo to describe the green point, and using this as a prompt ZeroKey predicts the blue point. We show that salient points, given by the Schelling Points paper [10], are more easily describable and consistently retrievable than arbitrary points. Arbitrary points can cause ZeroKey to not find any points at all.

To illustrate this idea, we analyze ZeroKey's ability to generate salient points that align with human-selected Schelling points. Using a dataset [10] that captures human-selected Schelling points, we prompt ZeroKey with natural language descriptions provided by ChatGPT, simulating natural language descriptions that humans might use. This aims to replicate the human process of identifying and naming keypoints based on visual and linguistic salience. An illustration and the detailed settings can be found in the supplementary material. Our findings show that ZeroKey's generated points align with high-density regions reported by Chen et al. [10], suggesting that its 3D understanding of shape corresponds, to some extent, with human perception.

5.2. Point Describability and Consistency

During our experiments, we observed that certain keypoints were more difficult to retrieve than others. This was raised when performing our quantitative analysis on the KeypointNet dataset [54], which provides ground-truth keypoints for various 3D models. The assignment of a concise and descriptive text prompt to each ground truth point proved to be part of the challenge. We noticed a correlation: when a point was easy to name succinctly by a human observer, ZeroKey was able to retrieve it more effectively. Based on this intuition, we propose to study how describable different points are and how consistently ZeroKey can retrieve them.

To quantify how "describable" a point is, we leverage the distributions reported in the Schelling points paper [10]. Points with high-density values in the distributions reported by [10] are considered more describable due to their inherent semantic significance. We refer to such points as salient points. The evaluation algorithm is described in the supplementary material.

The qualitative results of our evaluation are showcased in Fig. 4. We ask Molmo to describe the green dot, using a simple prompt. We then give the prompt generated by Molmo to Zerokey to predict a 3D location of the original

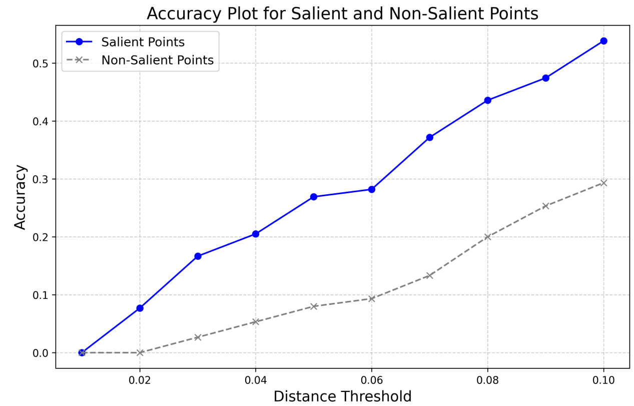


Figure 5. We show through a quantitative study that "salient" points are retrieved with higher accuracy than "non-salient" ones, regardless of the distance threshold used.

point. For salient points, ZeroKey successfully leverages Molmo's prompt to relocate the queried point with high accuracy. However, for non-salient (random) points, retrieval is significantly more challenging.ZeroKey may identify similar object parts (e.g., an airplane wing) or even unrelated regions (e.g., an incorrect area on a teddy bear). Additionally, we observed a fail state where for some non-salient points Molmo was unable to give a good prompt, resulting in ZeroKey affirming that no points matching the prompt were present in the scene.

Fig. 5 presents the quantitative results on the whole Schelling points dataset, showing that retrieval accuracy is significantly higher for semantically meaningful points than for arbitrary ones. This suggests that ZeroKey, guided by Molmo's descriptions, is better at reasoning about 3D structure when a point carries strong linguistic significance. These findings show that Molmo's 3D understanding is closely tied to the describability of keypoints.

6. Experiments

6.1. Setup and Dataset

We evaluate our method using the KeypointNet dataset. Our evaluation strategy follows the approach of You et al. [55], which computes the Intersection over Union (IoU) between predicted keypoints and ground-truth keypoints from the KeypointNet dataset, using varying distance thresholds. We also stick to the same three classes of the dataset for evaluation: airplane, chair, and table. Please refer to our supplementary for the details.

6.2. Zero-Shot 3D Keypoint Detection Baseline

RedCircle. The Red Circle method [42] is one of the simplest and most straightforward ways to highlight points in the image based on a text query by randomly sampling points and highlighting them with a red circle and picking the one point with the highest CLIP similarity to the text

Method	IoU (in %)										
Distance Thresholds	0.001	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
HARRIS-3D [43]	0.15	0.76	2.19	3.96	6.16	8.88	11.91	15.13	18.63	22.10	25.69
SIFT-3D [38]	0.29	1.05	2.62	4.38	6.65	9.42	12.38	15.65	19.39	22.71	26.15
ISS [57]	0.49	1.10	2.69	4.67	7.27	10.14	13.29	16.56	20.02	23.58	27.20
USIP [26]	0.83	1.70	3.25	5.24	8.00	11.76	15.98	20.63	25.56	30.20	33.67
D3FEAT [6]	2.36	3.86	7.27	10.37	13.37	17.12	21.41	26.12	30.39	34.82	38.41
UKPGAN [55]	3.95	6.54	12.27	17.86	22.42	26.55	30.28	34.32	38.46	42.65	46.49
FSKD [5]	7.00	7.94	11.17	16.74	23.99	31.14	38.14	43.97	49.30	53.62	57.03
B2-3D [49]	<u>12.41</u>	<u>20.29</u>	<u>35.55</u>	<u>46.25</u>	<u>52.74</u>	<u>57.72</u>	61.61	64.19	66.56	68.55	70.57
PaliGemma 2 [45]	0.00	0.34	1.03	2.98	4.25	5.70	6.62	8.17	8.92	11.49	11.65
RedCircle [42]	0.21	0.34	0.64	1.16	1.90	3.05	4.81	7.55	11.06	14.92	18.50
GPT-4o	0.18	0.48	1.20	2.61	4.19	6.04	8.48	10.89	14.03	17.03	20.73
CLIP-DINOiser [50]	0.73	1.41	3.00	4.94	7.31	9.80	12.66	15.52	18.52	21.76	25.56
StablePoints [17]	3.66	5.80	9.94	13.32	16.58	19.91	23.48	27.26	31.28	34.89	38.22
ZeroKey (Ours)	**9.12**	**13.16**	**24.01**	**37.43**	**48.43**	**56.60**	**<u>62.06</u>**	**<u>67.53</u>**	**<u>71.74</u>**	**<u>75.75</u>**	**<u>79.43</u>**

Table 1. Comparison of IoU between the predicted and ground-truth keypoints from KeypointNet using different methods across various geodesic distance thresholds. The **bold** text indicates the best zero-shot methods, while the <u>underlined</u> text highlights the best methods.

query. We lift the prediction of this method to 3D using the same lifting procedure used in our method to compare 3D Zero-shot keypoint detection.

CLIP-DINOiser. The CLIP-DINOiser [50] combines the strengths of CLIP and DINO to perform zero-shot object localization and segmentation based on text query in images. To adapt the CLIP-DINOiser for zero-shot 3D keypoint detection, we first apply the method to each view of the 3D scene to obtain 2D keypoint predictions corresponding to the text prompt. For each image, the CLIP-DINOiser produces a heatmap indicating the regions of interest related to the query. We select the point with the highest activation in the heatmap as the 2D keypoint prediction in that view. We then lift these 2D keypoints to 3D using the same back-projection technique described in our method.

StablePoint. In addition to the supervised and few-shot methods, we also compared our approach with an unsupervised baseline called StablePoint [17]. This method learns keypoints by optimizing text embeddings from latent diffusion models. The main idea is to identify text embeddings that guide the generative model to consistently focus on compact regions within images, which are then used as keypoints. Unlike our approach, this baseline does not retrieve keypoint according to a text prompt and operates fully unsupervised. However, it does necessitate specifying the exact number of keypoints we wish to detect. In our experiments, we set the number of keypoints to match the number of ground-truth points in the KeypointNet dataset.

GPT-4o. In this Baseline, we replace our model with GPT-4o to observe variations in performance. Notably, we highlight that MLLMs trained solely on image-level tasks fail to provide useful signals for keypoint detection. In the GPT-4o version, we replace Molmo model in our method with the latest GPT-4o (2024-08-06) and evaluate its performance on point-level tasks. This comparison highlights the necessity of specialized training or architectural features that support fine-grained reasoning in visual contexts.

PaliGemma 2. The 3B Mix 224 variant of PaliGemma 2 is a versatile and lightweight MLLM based on the Gemma language model [45]. It processes both images and text as input and generates text as output. Designed for a range of tasks—including short video captioning, visual question answering, text reading, object detection, and object segmentation—PaliGemma offers flexible multimodal capabilities. In our work, we adapt PaliGemma's object segmentation functionality to identify points of interest. Given an image and a text prompt, PaliGemma outputs a segmentation mask around the relevant region. We then select the area overlapped with the bounding box prediction as the detected points and lift these 2D points to 3D using our method.

6.3. Quantitative and Qualitative Analysis

Our KeypointNet evaluation shows (see Table.1) that our *Zero-Shot* method **significantly** outperforms MLLM-based baselines (PaliGemma 2[45], GPT-4o, CLIP-DINOiser [50]) across all distance thresholds. This provides strong evidence for our claim that the pixel-level annotations used to train MLLMs can be leveraged to both extract and name salient keypoints on 3D models *without requiring any ground truth labels or supervision.* Furthermore, our method achieves IoU levels comparable to those of reference-based *Few-Shot* and fully supervised methods tailored for this dataset, such as B2-3D [49]. Notably, after a certain distance threshold, our approach surpasses even few-shot methods, despite being zero-shot. The drop in performance at smaller thresholds is expected, given the inherently semantic nature of our approach, which prioritizes high-level understanding over fine-grained localization.

Side-by-side comparisons between ground truth keypoints and our Zero-Shot predictions, a figure of GPT-4o fails to precisely locate the keypoint, and a comparison of our detected keypoints with those of other baselines is

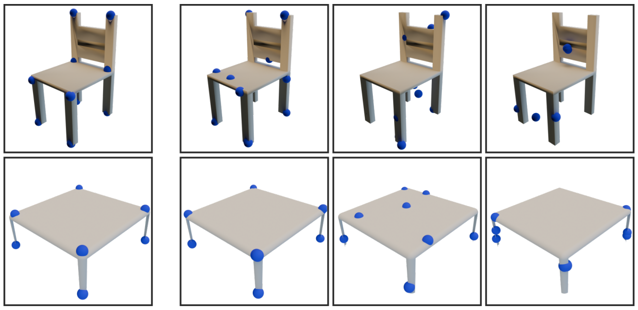


Figure 6. We compare against baselines CLIP-DINOiser and Red-Circle. While both baselines identify some prominent regions, they fall in accurately localizing keypoints according to the text prompt. In contrast, our method precisely locates keypoints.

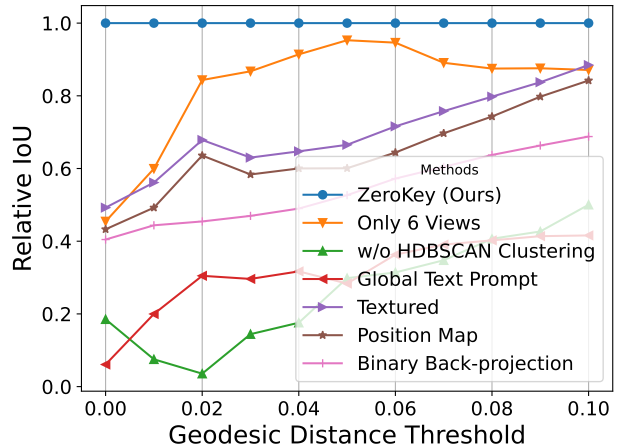


Figure 7. Comparison of the performance across different configurations: (blue) our original method; (red) results with a Global Text prompt; (orange, purple, brown) results using different rendering; (green) results without HDBSCAN clustering. This figure shows the impact of each modification on the overall performance.

shown in Fig. 6. We observe that our method consistently produces reasonable predictions, whereas MLLM-based baselines often fail to generate adequate results. These findings show the need for an MLLM backbone trained with point-level tasks for precise keypoint detection, underscore *the potential of point-level reasoning as a powerful task for MLLMs*, suggesting that it may induce a more generalizable understanding of vision and geometry compared to conventional image-level tasks like detection and segmentation.

6.4. Ablation Studies

In this ablation study, we modify different components of our method to gain deeper insights into the point-level reasoning abilities of MLLMs. Specifically, we compare the effects of various text prompts, experiment with different MLLM backbones, and evaluate the impact of aggregating information from different numbers of views.

Global Text prompt We evaluate the effectiveness of a point-specific text prompt. The original method utilizes prompts tailored to specific points in the image, providing precise guidance to the model. In the ablated version, we replace these point-specific text prompts with a global prompt: "Point to all salient points in this image." We then compare the detected set of keypoints with the ground truth to assess how this change affects the model's ability to localize specific points. This experiment helps us understand the importance of prompt specificity in guiding the MLLM's attention to particular regions of the image.

Aggregation ablation Our original method aggregates keypoint information from 26 viewpoints to improve accuracy. In the 6-views variant, we reduce the number of viewpoints to 6. In another variant, we compare the performance of direct average against HDBSCAN clustering for aggregating detections. By evaluating these different aggregation strategies, we aim to determine how the number of views and the method of combining information influence the model's ability to accurately identify keypoints.

The ablation results shown in Fig.7 illustrate the relative performance compared with ZeroKey. The ability to achieve 80% performance with only 6 views indicates that our method efficiently aggregates points. The poor performance of direct averaging suggests the necessity of HDBSCAN clustering. Binary Back-projection applies a constant weight of 1 for all points, rather than utilizing the soft voting weights, which significantly hinders its performance. Additionally, Replacing the shading with pointmap color and mesh texture from the dataset does not improve performance. This is because pointmap might be out-of-distribution of Molmo's training data. The poor quality of texture from KeypointNet also limit its effectiveness.

7. Conclusion and Future Work

In this paper, we presented a novel zero-shot approach for 3D keypoint detection by harnessing the capabilities of recent MLLMs trained with enhanced localized reasoning capabilities. Unlike traditional methods that rely heavily on annotated 3D datasets and extensive supervised training, our method leverages the pixel-level annotations inherent in MLLMs to both extract and name salient keypoints without any ground truth labels or supervision. Our evaluations demonstrate the efficacy of our approach and suggest that point-level reasoning is an effective way to endow MLLMs with a robust understanding of vision and geometry.

We believe that this work opens the door to entirely novel applications by providing a bridge between 3D data and MLLMs with *localized* reasoning. This can help solve downstream tasks such as shape manipulation, deformation, and processing, which typically require significant domain knowledge. Furthermore, this link can also help to enhance the power of MLLMs by providing feedback thanks to multiview consistency from 3D data, thus potentially helping to improve the robustness of their pixel-level understanding.

Acknowledgments. Parts of this work were supported by the ERC Consolidator Grant 101087347 (VEGA). The authors also gratefully acknowledge gifts from Ansys and Adobe Inc. This work was also supported by the KAUST Center of Excellence for Generative AI, under award number 5940, and the KAUST Ibn Rushd Postdoc Fellowship program.

References

[1] Ahmed Abdelreheem, Abdelrahman Eldesokey, Maks Ovsjanikov, and Peter Wonka. Zero-shot 3d shape correspondence. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3

[2] Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. Satr: Zero-shot semantic segmentation of 3d shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15166–15179, 2023. 3

[3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 1, 2, 3

[4] AI Anthropic. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 2024. 2

[5] Souhaib Attaiki and Maks Ovsjanikov. Ncp: Neural correspondence prior for effective unsupervised shape matching. *Advances in Neural Information Processing Systems*, 35:28842–28857, 2022. 2, 7

[6] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6359–6367, 2020. 7

[7] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 2

[8] Declan Campbell, Sunayana Rane, Tyler Giallanza, Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven M Frankland, Thomas L Griffiths, Jonathan D Cohen, et al. Understanding the limits of vision language models through the lens of the binding problem. *arXiv preprint arXiv:2411.00238*, 2024. 2

[9] Nenglun Chen, Lingjie Liu, Zhiming Cui, Runnan Chen, Duygu Ceylan, Changhe Tu, and Wenping Wang. Unsupervised learning of intrinsic structural representation points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9121–9130, 2020. 3

[10] Xiaobai Chen, Abulhair Saparov, Bill Pang, and Thomas Funkhouser. Schelling points on 3d surface meshes. *ACM Transactions on Graphics (TOG)*, 31(4):1–12, 2012. 3, 6

[11] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–468, 2018. 3

[12] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024. 2, 4, 5

[13] Clara Fernandez-Labrador, Ajad Chhatkuli, Danda Pani Paudel, Jose J Guerrero, Cédric Demonceaux, and Luc Van Gool. Unsupervised learning of category-specific symmetric 3d keypoints from point sets. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 546–563. Springer, 2020. 3

[14] Abdullah Hamdi, Silvio Giancola, and Bernard Ghanem. Mvtn: Multi-view transformation network for 3d shape recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11, 2021. 3

[15] Abdullah Hamdi, Silvio Giancola, and Bernard Ghanem. Voint cloud: Multi-view point cloud representation for 3d understanding. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[16] Abdullah Hamdi, Faisal AlZahrani, Silvio Giancola, and Bernard Ghanem. Mvtn: Learning multi-view transformations for 3d understanding. *International Journal of Computer Vision*, 2024. 3

[17] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Xingzhe He, Hossam Isack, Abhishek Kar, Helge Rhodin, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised keypoints from pretrained diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22820–22830, 2024. 3, 7

[18] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023. 2, 3, 4

[19] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4421–4430, 2019. 3

[20] Rui Huang, Songyou Peng, Ayca Takmaz, Federico Tombari, Marc Pollefeys, Shiji Song, Gao Huang, and Francis Engelmann. Segment3d: Learning fine-grained class-agnostic 3d segmentation without manual labels. *European Conference on Computer Vision (ECCV)*, 2024. 3

[21] Tomas Jakab, Richard Tucker, Ameesh Makadia, Jiajun Wu, Noah Snavely, and Angjoo Kanazawa. Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12783–12792, 2021. 3

[22] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[23] Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, and Siddhartha Chaudhuri. 3d shape segmentation with

projective convolutional networks. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3779–3788, 2017. 3

[24] 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. 3

[25] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3d semantic segmentation. In *European Conference on Computer Vision (ECCV)*, pages 518–535. Springer, 2020. 3

[26] Jiaxin Li and Gim Hee Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 361–370, 2019. 7

[27] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 1, 2, 3

[28] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 3

[29] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 1, 2, 3

[30] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. 3d-to-2d distillation for indoor scene parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4464–4474, 2021. 3

[31] Yecheng Lyu, Xinming Huang, and Ziming Zhang. Learning to segment 3d point clouds in 2d image space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12255–12264, 2020. 3

[32] Jinjie Mai, Abdullah Hamdi, Silvio Giancola, Chen Zhao, and Bernard Ghanem. Egoloc: Revisiting 3d object localization from egocentric videos with visual queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 45–57, 2023. 3

[33] Jinjie Mai, Wenxuan Zhu, Sara Rojas, Jesus Zarzar, Abdullah Hamdi, Guocheng Qian, Bing Li, Silvio Giancola, and Bernard Ghanem. Tracknerf: Bundle adjusting nerf from sparse and noisy views via feature tracks, 2024. 3

[34] OpenAI. Gpt-4 technical report, 2023. 1, 2, 4

[35] Wujian Peng, Sicheng Xie, Zuyao You, Shiyi Lan, and Zuxuan Wu. Synthesize diagnose and optimize: Towards fine-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13279–13288, 2024. 2

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 2, 3

[37] Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. *arXiv preprint arXiv:2407.06581*, 2024. 2

[38] Blaine Rister, Mark A Horowitz, and Daniel L Rubin. Volumetric image registration from invariant keypoints. *IEEE Transactions on Image Processing*, 26(10):4900–4910, 2017. 7

[39] Thomas C Schelling. *The Strategy of Conflict: with a new Preface by the Author*. Harvard university press, 1980. 5

[40] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. In *International Conference on Robotics and Automation (ICRA)*, 2023. 3

[41] Ruoxi Shi, Zhengrong Xue, Yang You, and Cewu Lu. Skeleton merger: an unsupervised aligned keypoint detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 43–52, 2021. 3

[42] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023. 3, 6, 7

[43] Ivan Sipiran and Benjamin Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27:963–976, 2011. 7

[44] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. *Advances in neural information processing systems*, 31, 2018. 3

[45] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy.

Gemma: Open models based on gemini research and technology, 2024. 7

[46] Brian H Wang, Wei-Lun Chao, Yan Wang, Bharath Hariharan, Kilian Q Weinberger, and Mark Campbell. Ldls: 3-d object segmentation through label diffusion from 2-d images. *IEEE Robotics and Automation Letters*, 4(3):2902–2909, 2019. 3

[47] Hanyu Wang, Jianwei Guo, Dong-Ming Yan, Weize Quan, and Xiaopeng Zhang. Learning 3d keypoint descriptors for non-rigid shape matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 3

[48] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International conference on machine learning*, pages 23318–23340. PMLR, 2022. 1

[49] Thomas Wimmer, Peter Wonka, and Maks Ovsjanikov. Back to 3d: Few-shot 3d keypoint detection with back-projected 2d features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4154–4164, 2024. 2, 4, 7

[50] Monika Wysoczanska, Oriane Siméoni, Michaël Ramamonjisoa, Andrei Bursuc, Tomasz Trzcinski, and Patrick Pérez. Clip-dinoiser: Teaching clip a few dino tricks for open-vocabulary semantic segmentation. In *Proc. ECCV*, 2024. 1, 7

[51] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer, 2024. 2, 4

[52] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1179–1189, 2023. 3

[53] Zi Jian Yew and Gim Hee Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *Proceedings of the European conference on computer vision (ECCV)*, pages 607–623, 2018. 3

[54] Yang You, Yujing Lou, Chengkun Li, Zhoujun Cheng, Liangwei Li, Lizhuang Ma, Cewu Lu, and Weiming Wang. Keypointnet: A large-scale 3d keypoint dataset aggregated from numerous human annotations. *arXiv preprint arXiv:2002.12687*, 2020. 3, 4, 6

[55] Yang You, Wenhai Liu, Yanjie Ze, Yong-Lu Li, Weiming Wang, and Cewu Lu. Ukpgan: A general self-supervised keypoint detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2, 6, 7

[56] Chenhui Zhang and Sherrie Wang. Good at captioning bad at counting: Benchmarking gpt-4v on earth observation data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 7839–7849, 2024. 2

[57] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 689–696. IEEE, 2009. 7

[58] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023. 2, 4

[59] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2, 3