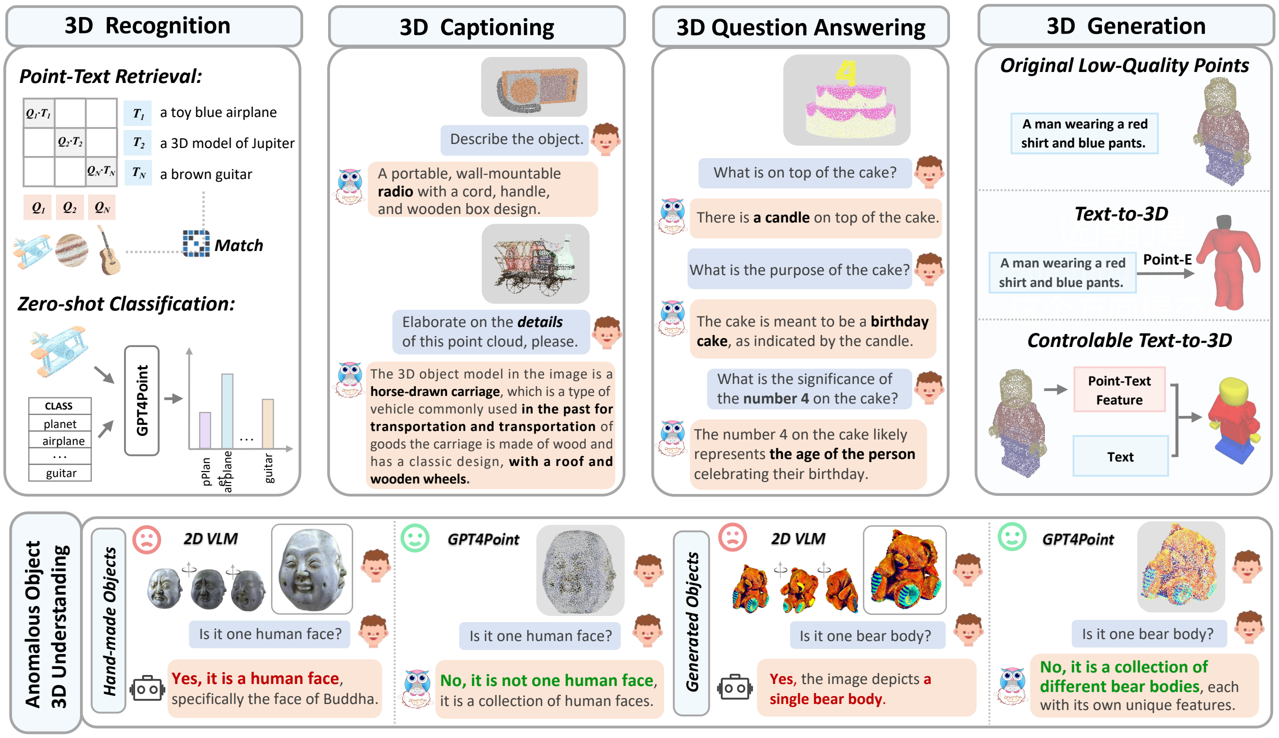
GPT4Point: A Unified Framework for Point-Language Understanding and Generation

Zhangyang Qi[1*] Ye Fang[3,5*] Zeyi Sun[4,5*]
Xiaoyang Wu[1] Tong Wu[2] Jiaqi Wang[5†] Dahua Lin[2,5] Hengshuang Zhao[1†]
[1]The University of Hong Kong [2]The Chinese University of Hong Kong
[3]Fudan University [4]Shanghai Jiao Tong University [5]Shanghai Artificial Intelligence Lab
https://gpt4point.github.io * equal contribution † corresponding author



Describe the object.

A portable, wall-mountable **radio** with a cord, handle, and wooden box design.

Elaborate on the ***details*** of this point cloud, please.

The 3D object model in the image is a **horse-drawn carriage**, which is a type of vehicle commonly used **in the past for transportation and transportation** of goods the carriage is made of wood and has a classic design, **with a roof and wooden wheels.**

What is on top of the cake?

There is **a candle** on top of the cake.

What is the purpose of the cake?

The cake is meant to be a **birthday cake**, as indicated by the candle.

What is the significance of the **number 4** on the cake?

The number 4 on the cake likely represents **the age of the person** celebrating their birthday.

Is it one human face?

Yes, it is a human face, specifically the face of Buddha.

Is it one human face?

No, it is not one human face, it is a collection of human faces.

Is it one bear body?

Yes, the image depicts **a single bear body**.

Is it one bear body?

No, it is a collection of different bear bodies, each with its own unique features.

Figure 1. **Task examples of GPT4Point**. It performs accurate 3D recognition, detailed captioning, precise Q&A, and high-quality controllable 3D generation. Additionally, GPT4Point excels in 3D anomalous object description, accurately assessing abnormal shapes like the multi-face object and the 3D generation failure case. It is a crucial ability in the assessment of generated 3D objects.

Abstract

Multimodal Large Language Models (MLLMs) have excelled in 2D image-text comprehension and image generation. Still, their understanding of the 3D world needs to be improved, limiting progress in 3D language understanding and generation. To solve this problem, we introduce GPT4Point, an innovative, groundbreaking point-language multimodal model explicitly designed for unified 3D object understanding and generation within the MLLM framework. GPT4Point, as a powerful 3D MLLM, can seamlessly execute point-text reference tasks such as point-cloud captioning and Q&A. Additionally, GPT4Point is equipped with advanced capabilities for controllable 3D generation, and it can get high-quality results through a low-quality point-text feature that maintains geometric shapes and colors. We develop Pyramid-XL, a point-language dataset annotation engine, to support the expansive needs of 3D object-text pairs. It constructs a large-scale database of over 1M objects of varied text granularity levels from the Objaverse-XL dataset, essential for training GPT4Point. A comprehensive benchmark has been proposed to evaluate 3D point-language understanding capabilities. In extensive evaluations, GPT4Point has demonstrated superior performance in understanding and generation.

1. Introduction

The recent Large Language Models (LLMs) [7, 37–39, 46, 51, 53, 55] have demonstrated remarkable advancements in natural language processing. Inspired by their powerful capabilities, researchers have also explored Multimodal LLMs (MLLMs) via adapting LLMs into various modalities like images [28, 32], audio [3, 16, 22] and videos [5, 67]. The proliferation of extensive image-text pair [4, 48] has crucially enabled 2D MLLMs *i.e.*, Vision Language Models (VLMs) to interpret images through textual representations. Concurrently, there is a growing trend in utilizing these multimodal models for guiding text-to-image generation [15, 49, 50, 60]. This represents a form of compression and reconstruction, exploring how to accurately recover and edit the input image using controllable image generation models. However, despite the impressive capabilities of MLLMs in handling multiple modalities, they still face significant limitations in understanding and accurately interpreting the 3D world, a critical need for various important downstream applications like intelligent robotics and augmented reality [9, 44].

Recent efforts to develop 3D MLLMs [21, 69] have notable limitations. Some [21, 69] prioritize the overall scene and focus primarily on the spatial coordinates of objects, often neglecting the geometric details of individual objects. This can lead to a limited understanding of the 3D world. Meanwhile, these methods generally convert 2D image features into 3D representations [21], which leads to a substantial loss of geometric accuracy. 3D geometry information is essential for understanding. As shown at the bottom of Fig. 1, the VLM fails to recognize the four-sided face object while our GPT4Point can figure out the anomalies. Concurrent works focusing on utilizing 3D features directly exhibit notable limitations. PointBind [18] exhibits a deficiency in training and demonstrates restricted text referencing abilities due to the limited dataset. On the other hand, PointLLM [57] necessitates the training of the corresponding Language Model (LLM) component and does not possess the capability to expand into text generation.

We present GPT4Point, a novel unified framework for point-language understanding and generation. GPT4Point introduces the 3D object MLLM, a groundbreaking language model that fully utilizes point clouds to perform various point-text tasks, as shown in Fig. 1. We use a Bert-based Point-QFormer for point-text feature alignment. Aligned features are separately input into the LLMs for text inference tasks and Diffusion for 3D object generation tasks. It is worth noting that, given a low-quality point cloud feature as a condition, GPT4Point can generate higher-quality results while maintaining the geometric shapes and colors by using point-text aligned features. This process can be called controllable text-to-3D, which becomes a milestone for 3D point textual editing.

To tackle the scarcity of object point-language data [52], we leverage the Objaverse-XL dataset [10, 11] to develop an automated, effective data annotation engine Pyramid-XL. It employs Vision Language Models (VLMs) for generating text annotations. Pyramid-XL solves the problem of VLMs needing to understand multi-view images directly. By synthesizing captions from multi-views obtained by the VLMs, the text annotation is stratified into three hierarchical levels, ranging from low to high, ultimately leading to precise annotations. Apart from the data engine, we establish an object point-text benchmark for assessing point multimodal model capabilities in recognition and text inference tasks, such as 3D object point cloud captioning and Q&A. This benchmark also provides a critical standard for evaluating 3D object generation, while current assessments often rely on qualitative judgments from rendered images without a direct evaluation in 3D space [43]. Only relying on rendering images may lead to misunderstanding; for instance, in the bottom right of Fig. 1, a failure case produced by 3D generation (a bear has two bodies) makes 2D VLMs, and even humans fail to recognize its anomaly. However, our model can identify anomalies quickly.

Our paper makes three major contributions:

- We present the unified framework for point-language understanding and generation **GPT4Point**, including the 3D MLLM for point-text tasks and controlled 3D generation.
- Introducing the automated point-language dataset annotation engine **Pyramid-XL** based on Objaverse-XL, currently encompassing 1M pairs of varying coarseness and can be extended cost-effectively.
- Establishing a novel object-level point cloud benchmark with comprehensive evaluation metrics for 3D point cloud language tasks. This benchmark thoroughly assesses models' understanding capabilities and facilitates the evaluation of generated 3D objects.

2. Related Work

Multimodal large language models (MLLMs). Large Language Models (LLMs) have demonstrated robust capabilities in language comprehension, reasoning, and generalization [7, 37–39, 46, 51, 53, 55]. Building upon this, Multimodal Large Language Models (MLLMs) extend these reasoning skills to additional modalities such as image [13, 14, 17, 64, 66, 68], audio [3, 16, 22], and video [5, 29]. Typically, MLLMs align target features with textual features and then integrate them with LLMs for various text inference tasks. Some train the whole architecture from scratch [23, 42], and others [1, 8, 26, 28, 32] utilize pre-trained LLMs. In 3D MLLMs, existing models either rely on 2D image information [21, 69] or align low-quality textual phrases with points [19, 57]. To solve these prob-

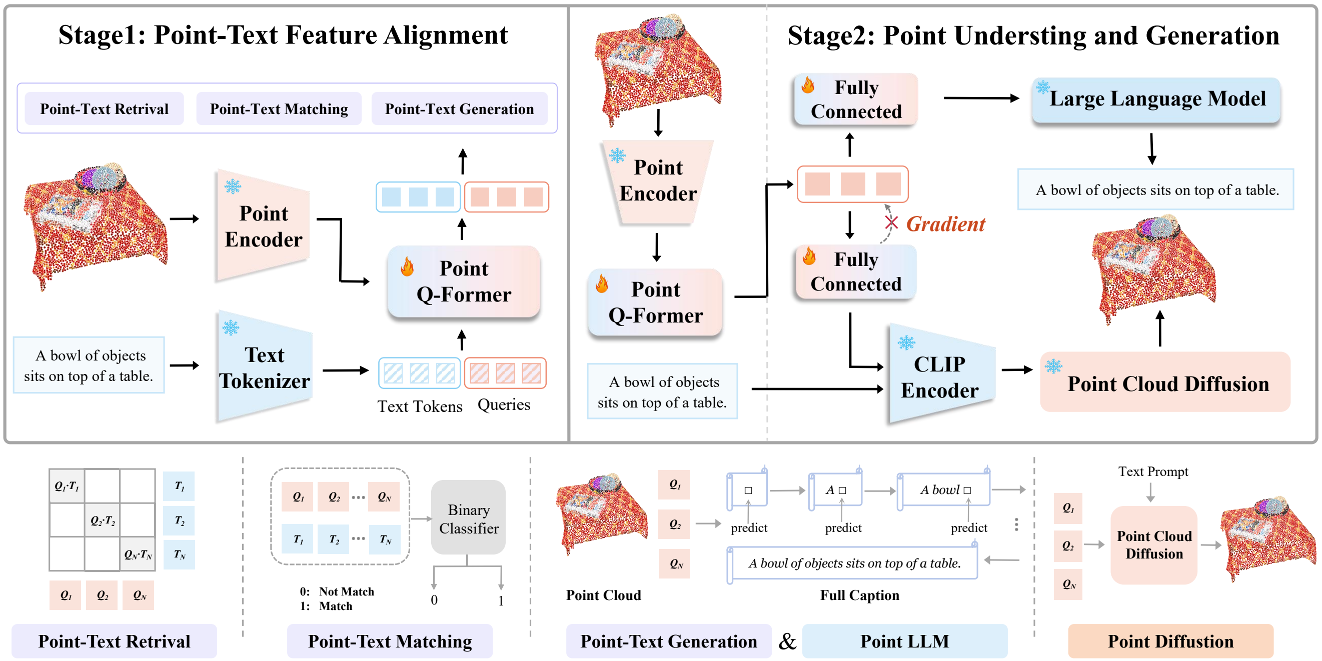


Figure 2. **The model architecture of GPT4Point for training.** In Stage 1, we employ a Bert-based [12] Point-Q-Former for point-text feature alignment through three point-text tasks. Then, in Stage 2, an LLM is appended to train the model's text inference capabilities. A Point Cloud Diffusion is attached separately to train controlled text-to-3D generation, which keeps the geometry shape and colors.

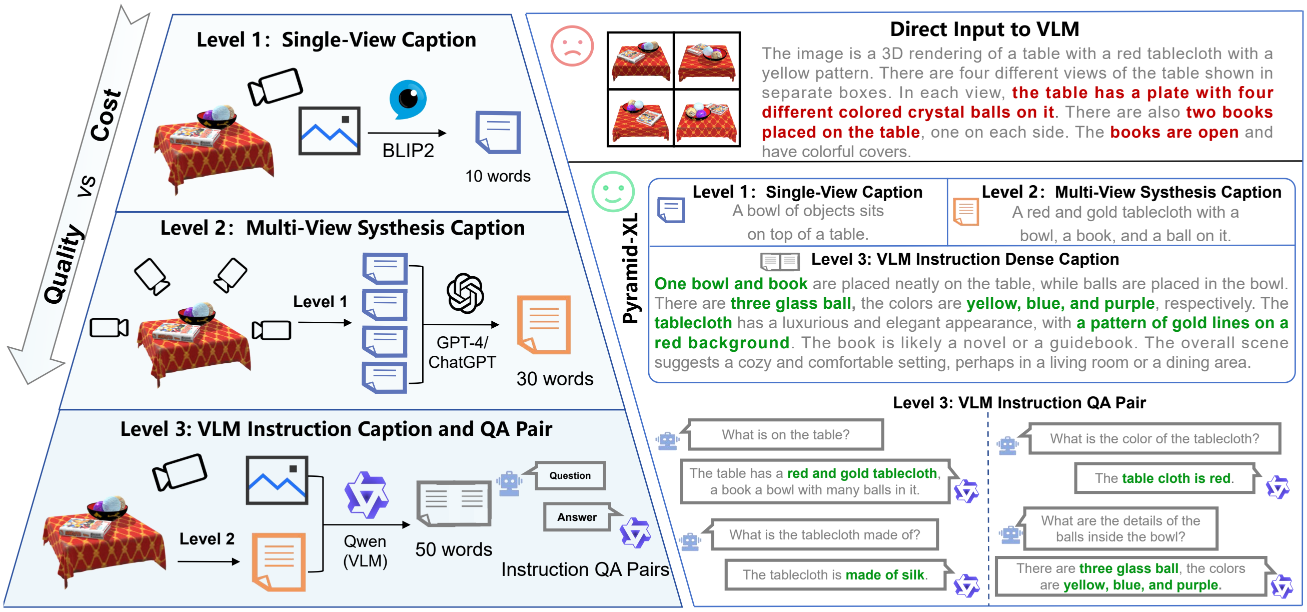
lems, we introduce a novel 3D MLLM designed for diverse point-text tasks. Our model, featuring a Point Q-Former based on Bert [12], aligns two domain features and integrates an LLM for text-based reasoning tasks, advancing 3D multimodal understanding.

Language-driven 3D object understanding. 3D point cloud multimodal models encompass a broad spectrum, generally categorized into those focusing on the entire scene containing multiple objects and those focusing on individual objects. The former emphasizes the relative positions of objects in the scene rather than their geometric shapes. Here, we primarily focus on the latter. In a self-supervised way, robust backbones like PointBert [61] for object points have been obtained [40, 61]. Then, point cloud language pretraining attempts to align the point cloud and text modalities. Some methods [24, 63] try to convert point clouds to depth images for alignment with text using CLIP [45]. Tri-modal approaches such as ULIP [18, 58, 59, 66] integrate point cloud, text, and image data. However, these methods all exclusively use 2D images explicitly or implicitly. Our work differs by directly aligning 3D point-text modalities, removing the dependency on image data.

Text-to-3D generation. Text-to-image generation models have experienced significant advancements recently [47, 62], yet text-to-3D models face challenges due to limited 3D data availability. Current approaches often rely on optimizing Neural Radiance Fields (NeRF) representation [35] with Score-Distillation-Sampling (SDS) loss [43]. While these optimization-based methods [6, 30, 43, 54] still fall short in robustness, speed, and generalization. Alternatively, Point-E [36] and Shap-E [25] employ feed-forward 3D generative models trained on large, undisclosed 3D datasets, offering better generalization and faster processing. However, these models often produce random, uncontrollable outputs with low-quality textures. To solve these limitations, we leverage point-text features to enhance the controllability of feed-forward models. This approach uses a low-quality point-text feature as a condition that allows for maintaining specific shapes and colors, thereby enabling the generation of higher-quality 3D objects.

3. Methods

This section provides an overview of our data text annotation engine and model architecture. In Sec. 3.1, we introduce **Pyramid-XL**, our point-language dataset annotation engine, discussing its design, function, and the progression from low-quality descriptions to ultimately precise and detailed ones. Then, in Sec. 3.2, we delve into GPT4Point's architecture, explaining how to align point and text and demonstrating how LLM and point diffusion models contribute to unified understanding and generation.


Quality
vs
Cost
Level 1 : Single-View Caption
BLIP2
10 words
Level 2 : Multi-View Systhesis Caption
Level 1
GPT-4/
ChatGPT
30 words
Level 3: VLM Instruction Caption and QA Pair
Level 2
Qwen
(VLM)
50 words
Question
Answer
Instruction QA Pairs
Direct Input to VLM
Pyramid-XL
Level 1 : Single-View Caption
Level 2 : Multi-View Systhesis Caption
Level 3: VLM Instruction Dense Caption
Level 3: VLM Instruction QA Pair

The image is a 3D rendering of a table with a red tablecloth with a yellow pattern. There are four different views of the table shown in separate boxes. In each view, **the table has a plate with four different colored crystal balls on it**. There are also **two books placed on the table**, one on each side. The **books are open** and have colorful covers.

A bowl of objects sits on top of a table.

A red and gold tablecloth with a bowl, a book, and a ball on it.

One bowl and book are placed neatly on the table, while balls are placed in the bowl. There are **three glass ball**, the colors are **yellow, blue, and purple**, respectively. The **tablecloth** has a luxurious and elegant appearance, with **a pattern of gold lines on a red background**. The book is likely a novel or a guidebook. The overall scene suggests a cozy and comfortable setting, perhaps in a living room or a dining area.

What is on the table?

The table has a **red and gold tablecloth**, a book a bowl with many balls in it.

What is the tablecloth made of?

The tablecloth is **made of silk**.

What is the color of the tablecloth?

The **table cloth is red**.

What are the details of the balls inside the bowl?

There are **three glass ball**, the colors are **yellow, blue, and purple**.

Figure 3. **Pyramid-XL: An automated point-text annotation engine.** Directly inputting images into VLMs yields unsatisfactory results. We propose a progressive annotation approach with 3 levels of granularity, leveraging results from the previous level for precise outcomes.

3.1. Point-Language Dataset Annotation Engine

The public release of the large-scale Objaverse dataset [11] and its successor Objaverse-XL [10] includes 800K and 10M objects, respectively, providing vast 3D object data. However, these objects lack corresponding text descriptions. We plan to use the rendered images of the objects as input and obtain textual descriptions through a trained Vision Language Model (VLM). However, direct input of multi-view images into the VLM does not enable it to understand their 3D structure and give precise descriptions, as shown in the top right of Fig. 3. Hence, Pyramid-XL employs a hierarchical pipeline, evolving from initial low-quality descriptions to achieve precise and detailed results.

Pyramid-XL

Single-View Caption (Level 1): We use the primary VLM model BLIP-2 [28] to generate concise descriptions, approximately ten words in length, from a single-view rendered image.
Multi-View Caption (Level 2): This level synthesizes multiple Level 1 descriptions by GPT-4 [38] to create comprehensive multi-view captions, which have approximately thirty words.
VLM Instruction Caption and QA Pair (Level 3): Utilizing the view with the highest CLIP score, selected from textual descriptions, we engage the advanced VLM to produce detailed dense captions and a corresponding QA dataset.

In terms of scale, Pyramid-XL is employed to annotate over 1M objects with Level 1 captions, 660K objects with Level 2 captions (same as Cap3D [34]), and 70K objects with Dense Captions including QA data. To assess the impact of text granularity on training, we designate the 1M Level 1 captions as the training dataset, while a smaller set of detailed Level 3 data is used for instruction tuning. This methodology mirrors practices in the vision field, where models are initially trained on large volumes of coarser data and finetuned on more detailed data from specialized domains. Detailed experimental results of this approach are presented in Sec. 5.3.

3.2. Model Architecture

GPT4Point consists of two stages, as illustrated in Fig. 2. In Stage 1, we focus on point-text alignment using the Point-QFormer, a Bert-based structure similar to the Q-Former in BLIP-2 [28]. This stage involves supervision through three tasks related to recognition and text reasoning. In Stage 2, only the point cloud is input into the point encoder and Point-QFormer to obtain aligned features, divided into the LLM Branch and the Diffusion Branch. These branches supervise text comprehension and object generation tasks, respectively.

Stage 1: point-text feature alignment. Given a point cloud $\boldsymbol{P} \in \mathbb{R}^{N \times 6}$, where each point is represented by six dimensions (XYZ coordinates and RGB color values), the initial stage of training focuses on feature extraction. The point encoder $\mathcal{E}$ processes the point cloud to yield the point cloud feature token $\boldsymbol{T}_1^p = \mathcal{E}(\boldsymbol{P})$. Concurrently, the input text is tokenized via the Point Q-Former's text tokenizer, resulting in the text feature token $\boldsymbol{T}_1^t$. These tokens, $\boldsymbol{T}_1^p$ and $\boldsymbol{T}_1^t$, are then utilized as inputs for the Point Q-Former $\mathcal{F}_Q$, facilitating the fusion of point cloud and textual data. We jointly optimize three training objectives: Point-Text Contrast (PTC) and Point-Text Matching (PTM), both recognition tasks, along with Point Caption Generation (PTG), a text inference task designed for aligning point clouds with textual data. The formulas are as follows:

$$\mathcal{L}_1 = \mathcal{F}_Q\left(\boldsymbol{T}_1^p, \boldsymbol{T}_1^t\right) = \mathcal{F}_Q\left(\mathcal{E}\left(\boldsymbol{P}\right), \boldsymbol{T}_1^t\right) \tag{1}$$

Here, $\mathcal{L}_1$ represents the loss for three tasks, and we have set the weight ratios between them all to 1. In the final layer of $\mathcal{E}$, a fully connected layer maintains consistency between the dimensions of $\boldsymbol{T}_1^p$ and $\boldsymbol{T}_1^t$.

Stage 2: point understanding and generation. After the point-text feature alignment, we proceed with understanding and generation tasks. We must only input the point cloud into the Point Encoder and Point Q-Former to obtain the aligned feature. A Large Language Model (LLM) is integrated with the Point Q-Former to help understand the task. The semantically integrated point cloud features are represented as $\boldsymbol{T}_2^P = \mathcal{F}_Q\left(\boldsymbol{T}_1^p\right) = \mathcal{F}_Q\left(\mathcal{E}\left(\boldsymbol{P}\right)\right)$. The textual feature tokens $\boldsymbol{T}_2^t$ are obtained from the LLM's tokenizer. The objective function is defined as follows:

$$\mathcal{L}_2 = \mathcal{F}_{LLM}\left(\boldsymbol{T}_2^p, \boldsymbol{T}_2^t\right) = \mathcal{F}_{LLM}\left(\mathcal{F}_Q\left(\mathcal{E}\left(\boldsymbol{P}\right)\right), \boldsymbol{T}_2^t\right) \tag{2}$$

$\mathcal{F}_Q$ indicates Point Q-former including a fully connected layer in its last layer to ensure consistency between the dimensions of $\boldsymbol{T}_2^p$ and $\boldsymbol{T}_2^t$. $\mathcal{L}_2$ represents the loss function from the Point Caption task alone.

For 3D object generation, we utilize the features obtained from low-quality point clouds via the Point Q-Former as conditions inputted into the text-to-3D framework. This process generates refined 3D objects that maintain consistency in shape and color with the original point cloud. A notable distinction from the LLM branch is that we have frozen point cloud diffusion and Point Q-Former. As shown in Fig. 2, we employ a single fully-connected layer to project the aligned features into the CLIP token embedding space, referred to as $\boldsymbol{T}_3^p$, and then concatenate these with the original text embeddings $\boldsymbol{T}_3^t$ using the CLIP tokenizer. The output from the CLIP text encoder, enriched with information from the original point cloud, is instrumental in enabling effective text-to-3D generation. The final output is achieved using Point E. This framework is inspired by BLIP-Diffusion [27] techniques used in subject-driven 2D generation. However, the critical distinction here from BLIP-Diffusion lies in how we concatenate the Clip text token and Q-Former feature. This difference may also stem from variations in the data volumes between 2D and 3D, which will be thoroughly examined in the appendix.

4. Benchmarks and Evaluation

Evaluating the performance of multimodal models presents significant challenges due to the need for more mature metrics to assess the quality of generated texts. For 3D objects, benchmarks primarily rely on human judgment or GPT-based assessments [57]. There are two key issues to consider in this context. Firstly, the evaluation process involves a certain degree of subjectivity. Identical results might receive varying scores, leading to an element of randomness. Secondly, each evaluation incurs time and monetary costs. In this section, we present the evaluation benchmark we have proposed, which is primarily designed to be objective, ensuring repeatability and verifiability. Sec. 4.1 outlines the composition of our test set. Sec. 4.2 addresses the evaluation of recognition capabilities, while Sec. 4.3 provides a detailed assessment of text inference abilities.

4.1. Composition of Test Set

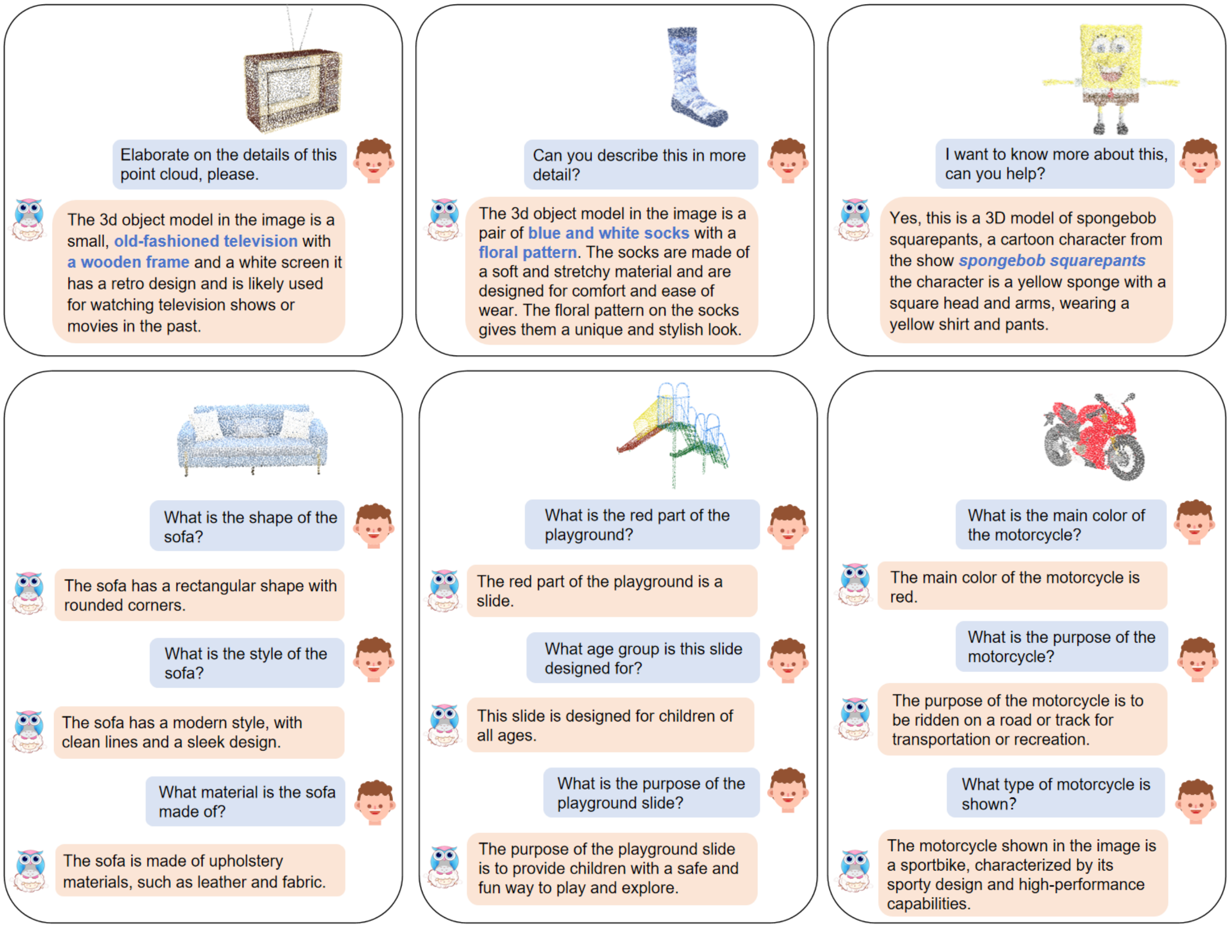
We leverage the Objaverse dataset [11], aligning it with LVIS categories [20], to create Objaverse-LVIS validation and test sets. In Objaverse-LVIS, we exclude scenes with complex settings, such as indoor houses or outdoor parks, focusing more on scenarios with single objects or combinations of multiple objects. We construct validation and test sets, each containing 1K objects. Compared to the PointLLM [57], which uses only 200 unfiltered objects as a test set, our more extensive set of 1K objects better measures the model's generalization capabilities. We initially use Pyramid-XL for textual descriptions to get initial annotations, followed by multiple rounds of expert manual revisions, ensuring comprehensive and accurate descriptions.

4.2. 3D Object Recognition

3D object recognition represents the classification capabilities of 3D multimodal models and the ability to match point cloud features with textual features. Objective measures, like accuracy, are typically used for evaluation.

Zero-shot point classification. Zero-shot point classification is considered a classic task in this domain. The widely used ModelNet40 dataset [56], which includes 2,468 objects across 40 categories, serves as a benchmark to evaluate a model's classification capabilities. In the multimodal context, the typical approach involves using the text *'a 3D model of [name]'* as input to match the point cloud modal features. The accuracy metric ACC@1, indicating the precision of top-1 rankings, best reflects the model's ability to match object categories accurately.

3D point-text retrieval. In 3D Point-Text Retrieval, we initially select 128 candidates based on point-text feature similarity and then re-rank these candidates using matching scores. Unlike classification tasks, which usually involve simple category names, the text can have more complex de-



Elaborate on the details of this point cloud, please.

The 3d object model in the image is a small, **old-fashioned television** with **a wooden frame** and a white screen it has a retro design and is likely used for watching television shows or movies in the past.

Can you describe this in more detail?

The 3d object model in the image is a pair of **blue and white socks** with a **floral pattern**. The socks are made of a soft and stretchy material and are designed for comfort and ease of wear. The floral pattern on the socks gives them a unique and stylish look.

I want to know more about this, can you help?

Yes, this is a 3D model of spongebob squarepants, a cartoon character from the show ***spongebob squarepants*** the character is a yellow sponge with a square head and arms, wearing a yellow shirt and pants.

What is the shape of the sofa?

The sofa has a rectangular shape with rounded corners.

What is the style of the sofa?

The sofa has a modern style, with clean lines and a sleek design.

What material is the sofa made of?

The sofa is made of upholstery materials, such as leather and fabric.

What is the red part of the playground?

The red part of the playground is a slide.

What age group is this slide designed for?

This slide is designed for children of all ages.

What is the purpose of the playground slide?

The purpose of the playground slide is to provide children with a safe and fun way to play and explore.

What is the main color of the motorcycle?

The main color of the motorcycle is red.

What is the purpose of the motorcycle?

The purpose of the motorcycle is to be ridden on a road or track for transportation or recreation.

What type of motorcycle is shown?

The motorcycle shown in the image is a sportbike, characterized by its sporty design and high-performance capabilities.

Figure 4. **Examples of text inference using the GPT4Point with ViT-g and OPT6.7B after Instruct Finetuning.** The table showcases its proficiency with point cloud input, excelling in tasks like detailed caption generation and point cloud-based question answering. This underscores our model's profound grasp of point cloud geometry and color, translating them into meaningful semantics.

scriptions. The evaluation metrics used are similar to those in image-text retrieval. We employ R1, R5, and R10 metrics to measure the accuracy of the top 1, 5, and 10 results in correctly matching points to text and vice versa.

4.3. 3D Object Text Inference

3D object text inference deeply represents the understanding capabilities regarding objects, including 3D object point cloud captioning and 3D point cloud question answering.

3D point cloud captioning. This task primarily evaluates the model's ability to provide an overall summary of a 3D object. The captions in the Objaverse-XL-LVIS caption test set are primarily within 30 words and accurately describe the object's geometry, color, and state. Moreover, we predominantly employ standard image description metrics, such as BLEU1, BLEU4, METEOR, ROUGE-L, and CIDEr [2, 31, 41] for evaluation.

3D point cloud question answering. In addition to point cloud captioning, 3D point cloud question answering explores object details through multiple rounds of dialogue. For instance, we can further explore the color or shape of specific parts of an object or even infer its simple usage. The curated Objaverse-XL-LVIS short QA 1K test set features concise, straightforward questions and answers, allowing us to calculate answer accuracy conveniently. Besides accuracy, we also use metrics from captioning to evaluate model performance. It is important to note that, for a fair comparison, we solely utilize zero-shot learning, meaning no fine-tuning is conducted on this kind of short QA dataset.

5. Experiments

5.1. Training Details

We configure our setup to process 8,192 input point clouds, utilizing Point-BERT [61] as the backbone. This transformer-based network excels in capturing geometric and semantic features of object point clouds. Moreover, the backbone is pretrained through retrieval tasks like ULIP-2 [59]. We employ OPT [65] and FlanT5 [46, 55] as Large Language Models (LLMs). For the training process, we adopt an initial learning rate of 1e-4, weight decay of 0.05,

Model	Input Data Type	ObjaverseXL-LVIS **Retrieval** (1K test set)						ModelNet40[56]
		Point → Text			Text → Point			Accuracy
		R@1	R@5	R@10	R@1	R@5	R@10	Acc@1
Image-Text Modal								
BLIP-2	Single-View Image (Mesh with Color)	17.56	41.16	52.82	16.72	40.2	52.56	35.62
InstructBLIP†		20.4	43.1	55.3	13.7	32.5	42.7	31.48
Point-Text Modal								
PointLLM(Vicuna-7B)†	Point Cloud (+Color)	-	-	-	-	-	-	41.33
GPT4Point		**32.2**	64	**81.3**	**89.7**	**98.1**	**98.9**	**43.90**

Table 1. **Point-Text Retrieval on the Objaversexl-LVIS test dataset and zero-shot 3D classification on ModelNet40**. Please note that † denotes Generative 3D object classification, which refers to the process of classifying 3D objects based on the generation of captions.

Model	#Trainable Params	ObjaverseXL-LVIS **Caption** (1K test set)					ObjaXL-LVIS **QA** (1K)		
		BLEU1	BLEU4	METEOR	ROUGE	CIDEr	**Acc**	BLEU1	ROUGE
Image-Text Modal									
BLIP-2 (OPT$_{2.7B}$)	188M	22.2	3.0	10.3	28.2	32.3	13.4	14.2	16.8
BLIP-2 (OPT$_{6.7B}$)	188M	24.9	4.1	11.5	30.0	44.2	15.4	15.1	18.3
InstructBLIP(Vicuna7B)	202M	25.5	4.3	11.6	30.7	47.2	15.9	16.2	20.1
Qwen-VL(Qwen-7B)	7.2B	27.1	4.9	13.1	31.3	63.8	18.2	19.5	24.4
Point-Text Modal									
PointLLM (Vicuna-13B)†	13.3B	26.2	4.9	11.9	31.3	50.9	23.4	22.3	26.2
GPT4Point (OPT$_{2.7B}$)	110M	28.9	6.0	13.2	33.9	68.4	22.1	23.4	25.3
GPT4Point (OPT$_{6.7B}$)	110M	31.5	**7.2**	13.8	35.4	**78.7**	27.1	26.2	30.4
GPT4Point (FLANT5$_{XL}$)	110M	**32.2**	**7.2**	**14.2**	**35.5**	78.0	**27.6**	**26.3**	**31.3**

Table 2. **3D Object Point Caption and Question Answer (QA) on the Objaversexl-LVIS 1K test dataset.** For the BLIP series, only fine-tuning of the Q-Former structure is required, whereas models like PointLLM need fine-tuning of the large language model.

batch size of 32, and the AdamW optimizer [33]. All hyper-parameters remain unchanged in both stages. The training process takes ten epochs for each stage on 8 A100 GPUs.

5.2. Evaluation and Diverse Tasks

We evaluate our model on the benchmark we proposed in Sec. 4, which includes 3D object recognition and 3D object text inference. Additionally, we demonstrate the model's capability for controllable text-to-3D generation.

3D object recognition. Recognition capabilities are shown in Tab. 1, with zero-shot classification results on the right side. Our approach demonstrates superior performance, outperforming the Vision Language Model(VLM) Instruct-BLIP [8] by 12.42 points and surpassing PointLLM [57] by 2.57 points. PointLLM employs a generative approach to generate text results by prompting, limiting its direct recognition capabilities. The results for 3D point-text retrieval are shown on the left side. Our GPT4Point model outperformed other VLMs [1, 8, 28]. The results quantitatively highlight the challenges of single-viewpoint 3D object occlusions and biases, emphasizing our approach's advantages over traditional image-text models.

3D object text inference. Model's text inference capabilities are displayed in Tab. 2. On the left, 3D object point cloud captioning results confirm GPT4Point's superiority over pre-trained VLMs and PointLLM. Notably, the Point Q-Former structure allows the freezing of the LLM, significantly reducing training parameters. The results for 3D point cloud Q&A on the right side show that GPT4Point achieved the best zero-shot accuracy, surpassing Instruct-BLIP [8] by 11.7 points and outperforming PointLLM [57] by 4.2 points. Alongside quantitative results, Fig. 4 qualitatively demonstrates its detailed answers and multi-turn dialogue capabilities, with more examples in the appendix.

Controllable text-to-3D object generation. Here, we showcase the generative capabilities of our model. Given features of low-quality point clouds and textual descriptions, we can generate corresponding higher-quality point clouds, making text-to-3D more controllable. Fig. 6 displays experimental results; we compare our point feature condition with text or single image condition in Point-E, demonstrating that aligning features using point cloud and textual information significantly improves guidance for point cloud generation. It is worth noticing that compared to a single-view image rendered from the original 3D model; our Point Q-former feature is in a better condition, containing richer information about the geometric shape and detailed color information of 3D objects. This is the first step

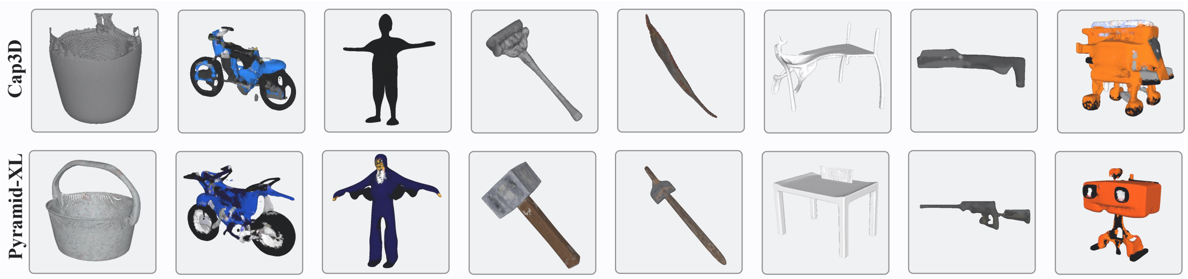


Figure 5. **Object generated from Point-E finetuned on Cap3D [34] and our Pyramid-XL** The first line shows Cap3D [34] finetuning results. In contrast, the second, using our Pyramid-XL Level 3 Dense Caption, outperforms Cap3D in geometry and color. This underscores the high quality of our text annotations.

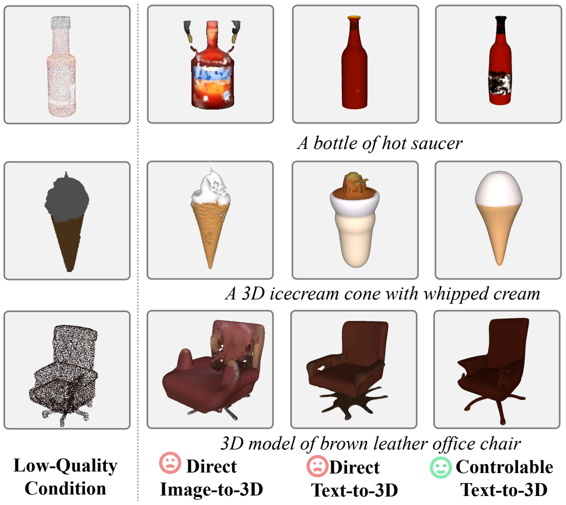


Figure 6. **Point-E generation result when conditioned on text, single image, and our Point-Q-former features**

towards the point cloud editing.

5.3. Assessing the Effectiveness of Pyramid-XL

In this section, we demonstrate the effectiveness of Pyramid-XL in obtaining high-quality point-text annotations. We focus on two tasks: finetuning Point-E [36] for 3D object generation using dense captions and utilizing annotations of varying granularities on the QA benchmark.

Finetune the Point-E with Level 3 Caption. We finetuned Point-E [36] base-40M text-vec model using 70K Level 3 VLM instruction captions from Pyramid-XL for 3D object generation. The results in Fig. 5 show significant improvements in geometric details and color fidelity in point clouds, especially in objects like baskets and Halloween costumes, compared to Cap3D [34].

Ablation study in model pretraining. Our ablation studies on Pyramid-XL, detailed in Tab. 4, investigated the impact of pretraining data scale and quality on model performance. The comparison between the first two rows indicates that using a large volume of coarse annotations boosts baseline, and Level 3 annotations lead to improvements.

Text-to-3D methods	Rendering Eval		User Study
	FID ↓	CLIP Score	Score(1-5)
Direct text-to-3D	34.7	74.9	3.98
Direct image-to-3D	32.6	75.3	3.67
Controllable text-to-3D	**31.6**	**76.2**	**4.03**

Table 3. Different 3D generation methods on the Cap3D, 2K test set. Our controllable text-to-3D achieved the best results.

Level of *Pyramid-XL*	3D Object QA	
	val	test
Level 2	22.3	22.1
Level 1 + Level 2	25.6	25.4
Level 1 + Level 2 + Level 3 (30%)	27.3	27.1
Level 1 + Level 2 + Level 3 (70%)	28.3	28.2
Level 1 + Level 2 + Level 3 (100%)	**28.5**	**28.4**

Table 4. **the impact of text granularity.** Levels 1, 2, and 3 represent coarse single-view annotation, multi-view annotation, and fine-grained annotation, respectively. After finetuning, we utilize the large language model OPT2.7B for pretraining and evaluation using ObjaverseXL-LVIS validation and test sets.

6. Conclusion

We introduce the innovative GPT4Point, a Unified Framework for point-language understanding and generation, including the 3D MLLM for point-text tasks and controlled text-to-3D generation based on low-quality point features. We develop Pyramid-XL, a point-language dataset annotation engine. This setup constructs a large-scale database over 1M objects of varied coarseness levels from the Objaverse-XL dataset. Furthermore, we establish an object-level point cloud benchmark with specific metrics for evaluating 3D point cloud-language tasks.

Acknowledgement. It is supported by National Natural Science Foundation of China (No. 62201484), HKU Startup Fund, HKU Seed Fund for Basic Research, National Key R&D Program of China (No. 2022ZD0160201) and Shanghai Artificial intelligence Laboratory.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv:2308.12966*, 2023. 2, 7

[2] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL Workshop*, 2005. 6

[3] Ching Chang, Wen-Chih Peng, and Tien-Fu Chen. Llm4ts: Two-stage fine-tuning for time-series forecasting with pretrained llms. *arXiv:2308.08469*, 2023. 2

[4] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 2

[5] Guo Chen, Yin-Dong Zheng, Jiahao Wang, Jilan Xu, Yifei Huang, Junting Pan, Yi Wang, Yali Wang, Yu Qiao, Tong Lu, et al. Videollm: Modeling video sequence with large language models. *arXiv:2305.13292*, 2023. 2

[6] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *ICCV*, 2023. 3

[7] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 2

[8] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2022. 2, 7

[9] Weinan Dai, Jinglei Tao, Xu Yan, Zhenyuan Feng, and Jinkun Chen. Addressing unintended bias in toxicity detection: An lstm and attention-based approach. In *ICAICA*, 2023. 2

[10] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. In *NeurIPS*, 2023. 2, 4

[11] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *CVPR*, 2023. 2, 4, 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 3

[13] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. In *ICML*, 2020. 2

[14] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv:2304.15010*, 2023. 2

[15] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv:2307.08041*, 2023. 2

[16] Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. Text-to-audio generation using instruction-tuned llm and latent diffusion model. In *ACMMM*, 2023. 2

[17] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *CVPR*, 2023. 2

[18] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv:2309.00615*, 2023. 2, 3

[19] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv:2309.00615*, 2023. 2

[20] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 5

[21] 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. In *NeurIPS*, 2023. 2

[22] Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, et al. Audiogpt: Understanding and generating speech, music, sound, and talking head. In *AAAI*, 2024. 2

[23] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. In *NeurIPS*, 2023. 2

[24] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *ICCV*, 2023. 3

[25] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv:2305.02463*, 2023. 3

[26] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv:2305.03726*, 2023. 2

[27] Dongxu Li, Junnan Li, and Steven CH Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. In *NeurIPS*, 2023. 5

[28] 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 *ICML*, 2023. 2, 4, 7

[29] Kunchang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv:2305.06355*, 2023. 2

[30] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023. 3

[31] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, 2004. 6

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 2

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 7

[34] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. In *NeurIPS*, 2023. 4, 8

[35] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 3

[36] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv:2212.08751*, 2022. 3, 8

[37] OpenAI. Chatgpt. `https://openai.com/blog/chatgpt`, 2022. 2

[38] OpenAI. GPT-4 technical report, 2023. 4

[39] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022. 2

[40] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *ECCV*, 2022. 3

[41] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6

[42] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv:2306.14824*, 2023. 2

[43] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023. 2, 3

[44] Zhangyang Qi, Jiaqi Wang, Xiaoyang Wu, and Hengshuang Zhao. Ocbev: Object-centric bev transformer for multi-view 3d object detection. In *3DV*, 2024. 2

[45] 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*, 2021. 3

[46] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. In *JMLR*, 2020. 2, 6

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 3

[48] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *CVPR*, 2023. 2

[49] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. In *ACL*, 2020. 2

[50] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. In *ICLR*, 2024. 2

[51] InternLM Team. Internlm: A multilingual language model with progressively enhanced capabilities, 2023. 2

[52] et al Tong Wu, Jiarui Zhang. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *CVPR*, 2023. 2

[53] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv:2302.13971*, 2023. 2

[54] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *NeurIPS*, 2023. 3

[55] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *ICLR*, 2021. 2, 6

[56] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015. 5, 7

[57] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. *arXiv:2308.16911*, 2023. 2, 5, 7

[58] 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 *CVPR*, 2023. 3

[59] Le Xue, Ning Yu, Shu Zhang, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *CVPR*, 2024. 3, 6

[60] Lili Yu, Bowen Shi, Ramakanth Pasunuru, Benjamin Muller, Olga Golovneva, Tianlu Wang, Arun Babu, Binh Tang, Brian Karrer, Shelly Sheynin, Candace Ross, Adam Polyak, Russell Howes, Vasu Sharma, Puxin Xu, Hovhannes Tamoyan, Oron Ashual, Uriel Singer, Shang-Wen Li, Susan Zhang, Richard James, Gargi Ghosh, Yaniv Taigman, Maryam Fazel-Zarandi, Asli Celikyilmaz, Luke Zettlemoyer, and Armen Aghajanyan. Scaling autoregressive multi-modal models: Pretraining and instruction tuning. *arXiv:2309.02591*, 2023. 2

[61] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *CVPR*, 2022. 3, 6

[62] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 3

[63] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *CVPR*, 2022. 3

[64] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. In *ICLR*, 2024. 2

[65] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv:2205.01068*, 2022. 6

[66] Yiyuan Zhang, Kaixiong Gong, Kaipeng Zhang, Hongsheng Li, Yu Qiao, Wanli Ouyang, and Xiangyu Yue. Meta-transformer: A unified framework for multimodal learning. *arXiv:2307.10802*, 2023. 2, 3

[67] Qihua Zhou, Song Guo, Jun Pan, Jiacheng Liang, Jingcai Guo, Zhenda Xu, and Jingren Zhou. Pass: Patch automatic skip scheme for efficient on-device video perception. *TPAMI*, 2024. 2

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

[69] Zhu Ziyu, Ma Xiaojian, Chen Yixin, Deng Zhidong, Huang Siyuan, and Li Qing. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *ICCV*, 2023. 2