VIT-LENS: Towards Omni-modal Representations

Weixian Lei[1,2] Yixiao Ge[2,3†] Kun Yi[2] Jianfeng Zhang[1] Difei Gao[1]
Dylan Sun[2] Yuying Ge[3] Ying Shan[2,3] Mike Zheng Shou[1†]
[†]Corresponding authors
[1]Show Lab, National University of Singapore [2]ARC Lab, Tencent PCG [3]Tencent AI Lab

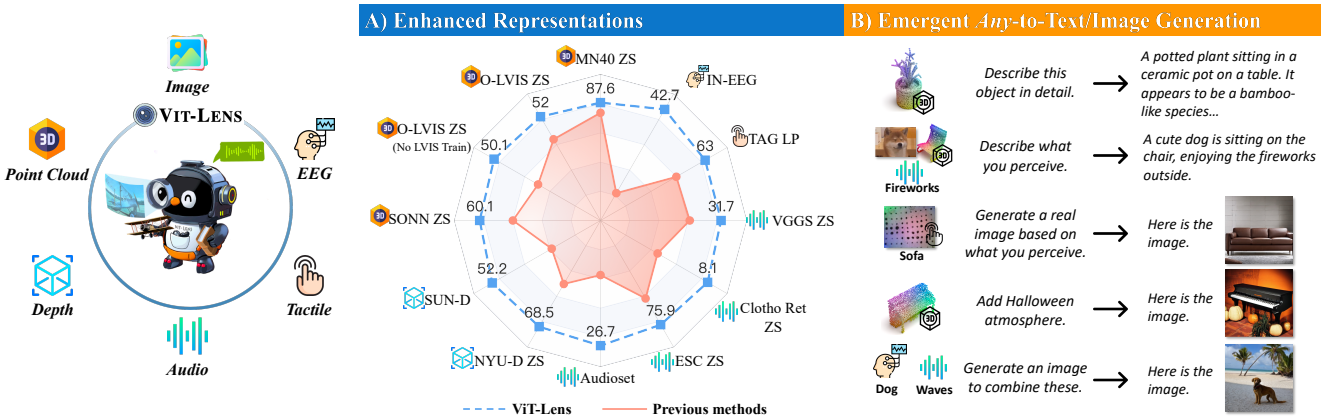


Figure 1. **VIT-LENS for omni-modal representation learning. A)** VIT-LENS consistently enhances the performance of understanding tasks, such as classification, zero-shot classification (ZS) and linear probing (LP), across 3D point cloud([54]), depth([32]), audio([32]), tactile([94]), and EEG([4]) modalities. The citations represent the compared previous methods. Further details in Sec. 4. **B)** By plugging VIT-LENS into multimodal foundation models, it enables emergent applications "out-of-the-box", including Any-modality Captioning/QA, Any-modality-to-Image Generation and text-guided Any-modality-to-Image editing, to name a few.

Abstract

Aiming to advance AI agents, large foundation models significantly improve reasoning and instruction execution, yet the current focus on vision and language neglects the potential of perceiving diverse modalities in open-world environments. However, the success of data-driven vision and language models is costly or even infeasible to be reproduced for rare modalities. In this paper, we present **VIT-LENS** *that facilitates efficient omni-modal representation learning by perceiving novel modalities with a pretrained-ViT and aligning them to a pre-defined space. Specifically, the modality-specific lens is tuned to project any-modal signals to an intermediate embedding space, which are then processed by a strong ViT with pre-trained visual knowledge. The encoded representations are optimized toward aligning with the modal-independent space, pre-defined by off-the-shelf foundation models.* VIT-LENS *provides a unified solution for representation learning of increasing modalities with two appealing advantages: (i) Unlocking the great potential of pretrained-ViTs to novel modalities effectively with efficient parameters and data regime; (ii) Enabling emergent downstream capabilities through modality alignment and shared ViT parameters. We tailor* VIT-LENS *to learn representations for 3D point cloud, depth, audio, tactile and EEG, and set new state-of-the-art results across various understanding tasks, such as zero-shot classification. By seamlessly integrating* VIT-LENS *into Multimodal Foundation Models, we enable Any-modality to Text and Image Generation in a zero-shot manner. Code and models are available at* https://github.com/TencentARC/ViT-Lens.

1. Introduction

Humans interact with the world through various sensory systems like vision, audition, touch, smell, and taste. To advance versatile AI agents, deep learning models need to replicate these human-like multi-sensory abilities and tackle varied user-specified tasks. For instance, visually interpret-

*This work extends [45] with added modalities and applications.

ing road signs to ensure our safe driving, listening to sirens to respond to emergency vehicles, and tactually assessing clothing fabric quality to offer shopping guidance. Among these applications, omni-modal representation learning has become a focal point, which enables comprehensive perception in open-world environments.

On the way to pursuing omni-modal AI agents, the research community has utilized large-scale web data to make substantial strides in language [6, 19, 55, 63, 74, 75, 84] and vision [5, 12, 20, 23, 24, 40, 42, 69, 76, 96]. Consequently, Multimodal Foundation Models (MFMs) [1, 11, 16, 21, 27, 28, 53, 105] that integrate vision representations with Large Language Models (LLMs) have made great progress in both vision-language comprehension and generation.

However, extending the success of unleashing LLMs to comprehend and interact with a broader array of modalities remains challenging. Despite recent initiatives [32, 87] in pursuing omni-modal intelligence, their capabilities on certain modalities are often constrained by limited data used in the training phase. In contrast to image, video, and text data, which are abundant on the internet, acquiring large-scale datasets for less common modalities can be non-trivial. This scarcity of data leads to sub-optimal models with poor generalization, particularly when encountering novel categories, thereby limiting their broader real-world applications.

In this work, we present a novel perspective. Given the exceptional generalization and transfer learning capabilities of the pretrained-ViTs [7, 23, 24, 65, 76], there is promise in adapting their inherent knowledge to comprehend novel modalities. This eliminates the necessity of collecting large-scale data to train models from scratch for each modality, which demands substantial time and resources. Recognizing the rich knowledge encoded in a pretrained-ViT, we conjecture that a pretrained-ViT is able to function as a multi-modal processor – it possesses the capacity to sense and comprehend a spectrum of modalities as it interprets images.

From this standpoint, we introduce **VIT-LENS**, which encodes the out-of-image modalities through a set of pretrained-ViT parameters, with the goal of maximizing the utilization of pretrained model weights and the knowledge they encapsulate. Specifically, VIT-LENS employs a modality-specific Lens along with a lightweight modality embedding module to transform input data into an intermediate space. Subsequently, a frozen pretrained-ViT is applied for further encoding. This approach enables the encoding of diverse modalities, aligning their features with the established features of anchor data, which can range from images, text to others, from off-the-shelf foundation models.

Our proposed method offers several advantages in advancing omni-modal representation learning: **(1) Parameters and data efficient approach.** Our method adopts a shared set of pretrained-ViT parameters across various modalities, enabling an efficient utilization of model parameters. Moreover, it efficiently enhances representations for less common modalities by leveraging the advanced ViT model, reducing the demand for extensive data collection. **(2) Emergent capability.** By training VIT-LENS with the ViT used in an off-the-shelf MFM, we can seamlessly obtain an Any-Modality MFM via VIT-LENS integration. The integrated model extends the original MFM's capabilities to various modalities, without any specific instruction tuning. For instance, without direct training for tactile data, the model broadens its image generation capability to include tactile-to-image generation. As a result, it can generate an image of a sofa upon receiving tactile signals indicating "leather".

We conducted comprehensive experiments across multiple modalities, extending beyond images and videos to encompass 3D point cloud, depth, audio, tactile, and EEG. This experiments were evaluated across 11 benchmarks. As is shown in Fig. 1A, VIT-LENS demonstrates state-of-the-art performance in 3D zero-shot classification. Particularly, when LVIS classes are excluded during training, VIT-LENS achieves an impressive zero-shot classification accuracy of 50.1% on Objaverse-LVIS [17], surpassing the prior SOTA by 11.0%. It consistently outperforms ImageBind [32] on depth and audio benchmarks, and surpasses previous works on tactile [94] and EEG [4] related tasks.

Beyond understanding tasks, we plug VIT-LENS into two recent MFMs, InstructBlip [16] and SEED [27, 28]. As illustrated in Fig. 1B, this empowers the MFMs to comprehend any modality in a zero-shot manner, making Any-Captioning, Any-QA, Any-to-Image Generation and text guided Any-to-Image editing right out of the box, all without the need for specific instruction tuning.

2. Related Work

Vision Language Pretraining: Advancements and Impacts. Recent advancements in vision-language pretraining, including models such as CLIP [76], ALIGN [42], CoCa [96], Flamingo [1], and LiT [98], have leveraged image-text pairs to achieve remarkable zero-shot performance on a wide range of vision and language tasks. Meanwhile, pretrained CLIP models have served as influential teachers and their joint embedding space has demonstrated efficacy in diverse zero-shot tasks such as segmentation [46], detection [35, 104], 3D shape understanding [54, 92, 93, 101, 106], 3D open-vocabulary segmentation [68], mesh animation [95], audio understanding [38] and more [100]. VIT-LENS extends these models' capacities to diverse modalities by integrating pretrained-ViT, enhancing its omni-modal understanding ability and enabling superior performance across various tasks and modalities.

Multimodal Learning. Previous studies explored joint training across multiple modalities in both supervised [26, 31, 50] and self-supervised settings [2, 33, 51, 59, 83]. Several approaches aim at aligning various modalities to CLIP for

multimodal zero-shot learning. AudioCLIP [38] adds audio to CLIP for zero-shot audio classification, while ImageBind [32] aligns six modalities to CLIP using paired image data. Besides, ONE-PEACE [87] introduces a unified encoder that is pretrained from scratch to align vision, language, and audio. Zhang *et al*. [102] pretrain a transformer with LAION-2B, following CLIP's methodology, for downstream supervised tasks across modalities. In contrast, VIT-LENS stands out by leveraging a pretrained-ViT to understand and unite diverse modalities without manual annotations. Its seamlessly integration with Multimodal Foundation Models (MFM) allows easy plug-and-play in emergent applications.

Multimodal Foundation Models. Recent advancements in Large Language Models (LLMs) [63, 84] have demonstrated remarkable language understanding and reasoning abilities. Afterwards, substantial efforts [1, 49, 52, 53, 105] have been directed towards enabling LLMs to perceive and interact with the visual world with the help of visual representation models. Similar paradigms enable LLMs to understand more modalities by algning the well-trained encoders of various modalities to the textual space of LLMs [36, 39, 80]. Beyond understanding tasks, recent works [27, 28, 81] empower LLMs with the ability to generate images, and NextGPT [88] extends the generative capabilities to encompass audio and video. Most of these Multimodal Foundation Models (MFMs) require specific instruction-following data within particular domains for training. In this study, we demonstrate that VIT-LENS can seamlessly integrate with an MFM without additional training, extending its capabilities to various modalities.

3. Method

Overview. VIT-LENS advances omni-modal representation learning by leveraging pretrained-ViT parameters to encode features for various modalities, utilizing the pre-existing knowledge in ViT. Specifically, a modality-specific encoder, composed of a modality embedding module, the modality-specific Lens, and the pretrained-ViT, is optimized to embed robust representations through the training objective of cross-modal alignment. For each modality, we consider its associated anchor modalities, like image and text, as reference points for learning. For alignment, we leverage foundation models, such as CLIP [76], to extract features from the anchor modalities. Our approach leverages the extensive knowledge embedded in both the foundation models and pretrained-ViT, providing a strong basis for representation learning for each modality. This compensates for the shortage of large-scale training data available for certain modalities. We illustrate our approach in Fig. 2.

3.1. Architecture

Foundation Models for alignment. In VIT-LENS, the new modalities are aligned to a unified feature space established by a robust foundation model. Various options exist for this model, ranging from language models [19, 55, 74, 75, 84], vision models [7, 24, 40, 65] to vision-language models [12, 47, 48, 76]. During training, we fix the foundation model's parameters and utilize it to encode features for the anchor data, which serves as supervision for feature alignment.

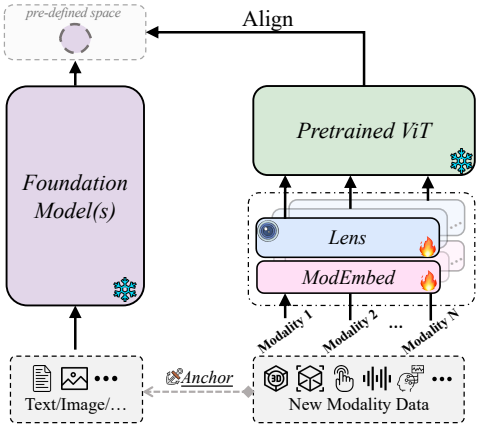


Figure 2. **Training Pipeline.** VIT-LENS extends the capabilities of a pretrained-ViT to diverse modalities. For each novel modality, it firstly employs a Modality Embedding (ModEmbed) and a Lens to learn mapping modality-specific data into an intermediate embedding space. It subsequently employs a set of pretrained-ViT layers to encode the feature. Finally, the output feature is aligned with the feature extracted from the anchor data (image, text, *etc*.) of the new modality using an off-the-shelf foundation model.

Modality Encoder. As is shown in Fig. 2, the modality encoder in VIT-LENS consists of a Modality Embedding Module (ModEmbed), a Lens and a set of pretrained-ViT layers. Due to the distinct characteristics of various modalities, raw signals may not match the pretrained-ViT input space. This mismatch can result in suboptimal performance, despite utilizing a powerful model. Therefore, we employ some heuristic designs: (1) *Obtain modality token embeddings:* for each modality, we adopt a specific tokenization scheme to transform raw input signals into token embeddings. (2) *Map modality token embeddings to the ViT input space:* the Lens learns to map the modality embeddings into a group of latent embeddings, thereby constructing the input for the pretrained-ViT. Subsequently, the latent embeddings are forwarded to frozen pretrained-ViT layers to obtain the final representation.

During training, the pretrained-ViT component remains frozen, and only the parameters of ModEmbed and Lens are updated. More details can be found in Supp.

Lens: Connecting Modalities to ViT. We introduce two variants of Lens to link modality token embeddings to ViT. We show their architectures in Fig. 3.

- **Self-attention blocks (S-Attn).** This variant involves a stack of self-attention layers [86] that transforms the input token embeddings into intermediate embeddings with

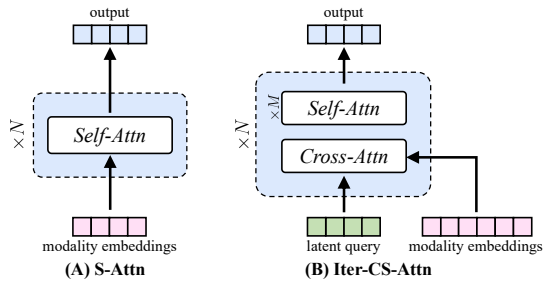


Figure 3. **Lens Architecture** used in VIT-LENS.

equal indices. We can potentially enhance this variant's capability by initializing it with pretrained weights from existing ViT layers. It suits modalities structured with image-like inputs, such as depth maps.

- **Iterative cross-self-attention blocks (Iter-CS-Attn).** This variant's basis block involves a cross-attention module coupled with a self-attention tower, inspired by [41]. It maps a latent array and input embeddings to a latent embedding of matching length within the input latent array. This manner condenses inputs of varied sized into a latent bottleneck, making it apt for lengthy input modalities like 3D point clouds. Similar architectures are employed in Vision-Language Models (VLMs) [1, 48] to extract visual information for Large Language Models (LLMs). Our innovation lies in utilizing this structure to map signals from diverse modalities into the pretrained-ViT's input space, enabling the ViT to understand modalities beyond images.

3.2. Training Objective

In this work, we study modalities of 3D, depth, audio, tactile and EEG, among which all the data samples are associated with text descriptions, image appearances, or both. We use the pretrained CLIP [12, 76] as the foundation model. By default, we employ pretrained layers of ViT in the foundation CLIP as part of the modality encoder. Following the approach in previous works [38, 54, 92, 93], we adopt multimodal contrastive learning for representation alignment.

We denote $X = \{x_1, \ldots, x_N\}$ as the collection of modality data to be learned, $\mathcal{A} = \{A_1, \ldots, A_M\}$ as the set of anchor modalities, a_n^m as the anchor data of x_n from modality A_m, $\mathbf{G}_A$ as the foundation model for anchor modality A, and $\mathbf{F}$ as the modality encoder to be learned. The contrastive loss for alignment is formulated as:

$$\mathcal{L} = -\frac{1}{2B|\mathcal{A}|}\sum_{i=1}^{B}\sum_{k=1}^{|\mathcal{A}|}\left(\log\frac{\exp(h_i^X \cdot h_i^{A_k}/\tau)}{\sum_j \exp(h_i^X \cdot h_j^{A_k}/\tau)} + \log\frac{\exp(h_i^{A_k} \cdot h_i^X/\tau)}{\sum_j \exp(h_i^{A_k} \cdot h_j^X/\tau)}\right),$$

where B is the batch size; τ is a learnable temperature; $h_i^X = \text{Norm}\left(\mathbf{F}(x_i)\right)$, $h_i^{A_k} = \text{Norm}\left(\mathbf{G}_{A_k}(a_i^k)\right)$ are normalized features of data x_i and its anchor data a_i^k from A_k.

3.3. Free Lunch for Multimodal Foundation Models

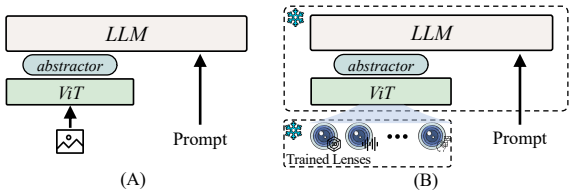


Figure 4. **Demonstration of integrating VIT-LENS to MFM.** (A) Original overall pipeline of MFM for vision; (B) Illustration of plugging well-trained Lenses of different modalities to MFM, **without** additional instruction-following training.

Plug VIT-LENS into MFM. Recent MFMs for vision [16, 27, 28, 52, 53, 105] enables LLMs to understand visual content. As shown in Fig. 4A, this process begins with the use of a frozen ViT to extract visual features. Subsequently, a well-trained abstractor module processes these features, constructing inputs that can be understood by the LLM.

By incorporating the ViT from MFM as part of the modality encoder and as the foundation model in VIT-LENS training, the yielded modality Lenses can be seamlessly integrated into the MFM for plug-and-play application, as depicted in Fig. 4B. In later experiments, we showcase the emergent abilities facilitated by this tuning-free adaptation.

4. Experiments

4.1. Experimental Setup

For this part, we describe the main experimental setup and provide full details in Supp.

Dataset	Task	#cls	Metric	#test
ModelNet40(MN40) [90]	3D shape cls	40	Acc	2,468
Objaverse-LVIS(O-LVIS) [17]	3D shape cls	1,156	Acc	46,832
ScanObjectNN(SONN) [85]	3D shape cls	15	Acc	581
SUN Depth-only(SUN-D) [78]	Scene cls	19	Acc	4,660
NYU-v2 Depth-only(NYU-D) [61]	Scene cls	10	Acc	654
Audioset Audio-only(AS-A) [29]	Audio cls	527	mAP	17,132[1]
ESC 5-folds(ESC) [70]	Audio cls	50	Acc	2,000
Clotho(Clotho) [22]	Retrieval	-	Recall	1,046
AudioCaps(ACaps) [44]	Retrieval	-	Recall	813[1]
VGGSound(VGGS) [9]	Audio cls	309	Acc	15,434[1]
Touch-and-go(TAG-M) [94]	Material cls	20	Acc	29,879
Touch-and-go(TAG-H/S) [94]	Hard/Soft cls	2	Acc	29,879
Touch-and-go(TAG-R/S) [94]	Rough/Smooth cls	2	Acc	8,085
ImageNet-EEG(IN-EEG) [79]	Visual Concept cls	40	Acc	1,997

Table 1. **Details of Downstream Datasets** across various modalities including 3D, depth, audio, tactile, and EEG. The evaluation is performed following feature alignment. The information presented includes the task type (classification/retrieval), the number of classes, the evaluation metric (Accuracy/mean Average Precision/Recall), and the quantity of test samples in each dataset.

[1]# test samples may differ from those used in previous work due to the unavailability of certain data.

	Top1	Top5
Trained on ULIP-ShapeNet [92]		
ULIP-PointNet++(ssg) [92]	55.7	75.7
ULIP-PointNet++(msg) [92]	58.4	78.2
ULIP-PointMLP [92]	61.5	80.7
ULIP-PointBERT [92]	60.4	84.0
VIT-LENS$_B$	65.4	92.7
VIT-LENS$_L$	**70.6**	**94.4**
Trained on ULIP2-Objaverse [93]		
ULIP2-PointNeXt [93]	49.0	79.7
ULIP2-PointBERT [93]	70.2	87.0
VIT-LENS$_B$	74.8	93.8
VIT-LENS$_L$	**80.6**	**95.8**

(a) Zero-shot 3D of classification on ModelNet40. Models are pretrained on triplets from ULIP-ShapeNet and ULIP2-Objaverse respectively.

	Objaverse-LVIS			ModelNet40			ScanObjectNN		
	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5
2D inference, no 3D training									
PointCLIP [101]	1.9	4.1	5.8	19.3	28.6	34.8	10.5	20.8	30.6
PointCLIP v2 [106]	4.7	9.5	12.9	63.6	77.9	85.0	42.1	63.3	74.5
Trained on OpenShape-Triplets (No LVIS) [54]									
ULIP-PointBERT [92]	21.4	38.1	46.0	71.4	84.4	89.2	46.0	66.1	76.4
OpenShape-SparseConv [54]	37.0	58.4	66.9	82.6	95.0	97.5	54.9	76.8	87.0
OpenShape-PointBERT [54]	39.1	60.8	68.9	85.3	96.2	97.4	47.2	72.4	84.7
VIT-LENS$_G$	**50.1**	**71.3**	**78.1**	**86.8**	**96.8**	**97.8**	**59.8**	**79.3**	**87.7**
Trained on OpenShape-Triplets [54]									
ULIP-PointBERT [92]	26.8	44.8	52.6	75.1	88.1	93.2	51.6	72.5	82.3
OpenShape-SparseConv [54]	43.4	64.8	72.4	83.4	95.6	97.8	56.7	78.9	88.6
OpenShape-PointBERT [54]	46.8	69.1	77.0	84.4	96.5	98.0	52.2	79.7	88.7
VIT-LENS$_G$	**52.0**	**73.3**	**79.9**	**87.6**	**96.6**	**98.4**	**60.1**	**81.0**	**90.3**

(b) Zero-shot 3D classification on Objaverse-LVIS, ModelNet40 and ScanObjectNN. Models are pretrained on OpenShape Triplets. "NO LVIS" denotes exclude the Objaverse-LVIS subset.

Table 2. Zero-shot 3D classification on downstream datasets, measured in accuracy(%).

Pretraining Datasets. Beyond image/video and text, we train VIT-LENS on a variety of modalities, including 3D point cloud, depth, audio, tactile, and EEG data. These datasets are anchored to text descriptions, images, or both for feature alignment.

For 3D point cloud experiments, we utilize a combination of ShapeNet [8], 3D-FUTURE [25], ABO [14], and Objaverse [17]. We incorporate rendered images and text captions from previous works, resulting in three pretraining datasets: ULIP-ShapeNet [92], ULIP-2-Objaverse [93], and OpenShape-Triplets [54]. Depth data is sourced from the SUN RGB-D dataset [78], utilizing paired image and scene labels for alignment. Audio data is obtained from the Audioset dataset [29], accompanied by associated video and text label metadata. Tactile data is sourced from the Touch-and-go dataset [94], featuring paired frame and material label text. Finally, EEG data from [79] is aligned with paired ImageNet image and text labels.

Evaluation on Downstream Understanding Tasks. We evaluate VIT-LENS across diverse modalities and protocols via a comprehensive set of downstream tasks. The primary datasets used for evaluation are summarized in Tab. 1.

Main Implementation Details. We use the pretrained vision and text encoders from OpenCLIP [12]. We apply different model sizes: VIT-LENS$_B$ based on ViT-B/16, VIT-LENS$_L$ based on ViT-L/14, and VIT-LENS$_G$ based on ViT-bigG/14.

For 3D point cloud data, we follow the baseline methods [54, 92] to uniformly sample 8,192 or 10,000 points and grouping them into sub-clouds through Farthest Point Sampling (FPS) followed by KNN grouping of neighboring points. For depth input, we follow [32] to use in-filled depth values and convert them to disparity for scale normalization. For audio data, we sample 5-second clips and extract a single frame randomly from the video clip if video serves as anchor data. The audio waveform is converted into a sequence of 128-dimensional log Mel filterbank features using a 25ms Hamming window every 10ms, following [34]. For tactile input, we use RGB data collected from GelSight [43]. For EEG signals, we employ 128-channel temporal sequences and use the frequency range of 5-95Hz following [4].

4.2. Results on Understanding Tasks

Zero-shot 3D Classification. We follow [54, 92, 93] to use (point cloud, image, text) triplets to train VIT-LENS. We conduct zero-shot classification on downstream benchmarks. The overall results can be found in Tab. 2. In particular, when pretrained on ULIP-ShapeNet or ULIP2-Objaverse, VIT-LENS outperforms ULIP with different 3D encoders [56, 72, 73, 97], as is shown in Tab. 2a.

We present the results of training on OpenShape-Triplets in Tab. 2b. To align with [54], we adopt VIT-LENS$_G$ and train on both "NO LVIS" (excluding all shapes from the Objaverse-LVIS subset) and the entire set for comparison. VIT-LENS outperforms models adopted in OpenShape [54]. Notably, VIT-LENS significantly improves the accuracy on the long-tail categories of Objaverse-LVIS, from 46.8% to 52.0%. Additionally, when trained on the NO LVIS subset, VIT-LENS achieves a top-1 accuracy of 50.1%. This performance beats ULIP by roughly 30% and surpasses OpenShape-PointBERT trained on the entire set by 3.3%, demonstrating the data-efficient merit of VIT-LENS. Regarding ModelNet40, VIT-LENS achieves an 87.4% accuracy, surpassing previous SOTA. Moreover, on ScannetObjectNN, containing challenging real scans with noise and occlusion, our method exhibits decent sim-to-real transfer ability. It achieves a 60.1% zero-shot accuracy without specific sim-to-real training, surpassing the previous SOTA.

Audio Classification and Retrieval. In our comparison presented in Tab. 3, VIT-LENS$_L$ consistently outperforms prior approaches in both audio classification and text-to-audio

	anchor	AudioSet mAP	VGGSound$^\diamond$ Top1	ESC$^\diamond$ Top1	Clotho$^\diamond$ R@1	Clotho$^\diamond$ R@10	AudioCaps$^\diamond$ R@1	AudioCaps$^\diamond$ R@10
AVFIC [60]	-	-	-	-	3.0	17.5	8.7	37.7
ImageBind-H [32]	I	17.6	27.8	66.9	6.0	28.4	9.3	42.3
VIT-LENS$_L$	I	23.1	28.2	69.2	6.8	29.6	12.2	48.7
AudioCLIP [38]	I+T	25.9	-	69.4	-	-	-	-
VIT-LENS$_L$	I+T	**26.7**	**31.7**	**75.9**	**8.1**	**31.2**	**14.4**	**54.9**
Prev. ZS SOTA	-	-	29.1/46.2* [89]	91.8 [87]	6.0	28.4 [32]	9.3	42.3 [32]

Table 3. Audio classification and retrieval on Audioset, VGGSound, ESC, Clotho and AudioCaps. $^\diamond$denotes zero-shot evaluation. Gray-out denotes using larger audio-text datasets in pretraining. *denotes using augmented captions for training.

	modality	R@1	R@5	R@10
MIL-NCE [57]	V	8.6	16.9	25.8
SupportSet [67]	V	10.4	22.2	30.0
AVFIC [60]	A+V	19.4	39.5	50.3
ImageBind-H [32]	A+V	36.8	61.8	70.0
VIT-LENS$_L$	A+V	**37.6**	**63.2**	**72.6**
Zero-shot SOTA [10]	V	49.3	68.3	73.9

Table 4. Video Retrieval on MSRVTT. V: use video; A+V: use audio and video. Gray-out means using video data in pretraining.

	anchor	NYU-D	SUN-D
Text Paired [32]	T*	41.9	25.4
ImageBind-H [32]	I	54.0	35.1
VIT-LENS$_L$	I	64.2	37.4
VIT-LENS$_L$	I+T	**68.5**	**52.2**
Supervised SOTA [31]	-	76.7	64.9

Table 5. Depth-only scene classification on NYU-D and SUN-D. *[32] rendered depth as grayscale images for direct testing. The supervised SOTA [31] used RGBD as input and extra training data.

	anchor	Material	H/S	R/S
ImageBind-B*	I	24.2	65.7	69.8
VIT-LENS$_B$	I	29.9	72.4	77.9
VIT-LENS$_L$	I	31.2	74.3	**78.2**
VIT-LENS$_L$	I+T	**65.8**	**74.7**	63.8
Linear Probing				
CMC [82, 94]	I	54.7	77.3	79.4
VIT-LENS$_B$	I	**63.0**	**92.0**	**85.1**

Table 6. Tactile classification on Touch-and-go. *denotes our implementation. H/S: Hard/Soft; R/S: Rough/Smooth.

	anchor	Val	Test
ImageBind-B*	I	17.3	18.4
DreamDiffusion-L$^\#$ [4]	I	20.4	19.2
VIT-LENS$_B$	I	24.6	25.3
VIT-LENS$_L$	I	29.3	29.2
VIT-LENS$_L$	I+T	**41.8**	**42.7**

Table 7. Visual concept classification on ImageNet-EEG. *denotes our implementation. $^\#$We use the released EEG encoder and paired text encoder for inference. We report results on Val and Test set.

retrieval tasks. When aligned to images (I), VIT-LENS$_L$ outperforms ImageBind based on Huge CLIP [12], and AVFIC [60], which leverages automatically mined audio-text pairs for alignment. When aligned to images and texts (I+T), VIT-LENS$_L$ demonstrates stronger performance and significantly outperforms AudioCLIP [38]. Although AudioCLIP uses a audio encoder pretrained with Audioset supervised classification, it falls behind VIT-LENS$_L$. Additionally, on zero-shot VGGSound classification, VIT-LENS$_L$ surpasses the SOTA [89] when class names are used as text supervision for alignment.

Audio and Video Retrieval. We use the MSR-VTT [91] benchmark to evaluate the text to audio and video retrieval performance, as presented in Tab. 4. We follow [32] to combine audio (A) and video (V) modalities. VIT-LENS outperforms several prior methods, even surpassing those that incorporate video data for training [57, 60, 67].

Depth-only Scene Classification. In Tab. 5, we present our results for depth-only classifications. VIT-LENS outperforms ImageBind across SUN-D and NYU-D. By using image and text as anchor data, VIT-LENS further improves the performance and narrows the gap with the supervised SOTA model [31] with extra training data.

Tactile Classification Tasks. Results for tactile tasks are displayed in Tab. 6. Across various tactile classification tasks like material, hard/soft, and rough/smooth classification, VIT-LENS$_B$ demonstrates superior performance compared to our implementation of ImageBind-B. Even trained with appearance or text labels for material, VIT-LENS can perform well on the hard/soft and rough/smooth classification tasks. This underscores the extensive knowledge transfer by CLIP during training. Furthermore, scaling up to a larger model and incorporating text during training can further boost the performance. In comparing the image-aligned VIT-LENS$_B$ with CMC [94] using linear probing, we observe significantly superior performance by VIT-LENS.

EEG Visual Concept Classification. Results in Tab. 7 show that VIT-LENS consistently outperforms our implemented ImageBind-B. Additionally, when compared to the EEG encoder from [4], which used more EEG data for MAE-style pretraining [40] and then aligned with the CLIP-L14 image encoder, VIT-LENS demonstrates superior performance.

4.3. Few-shot Linear Probing

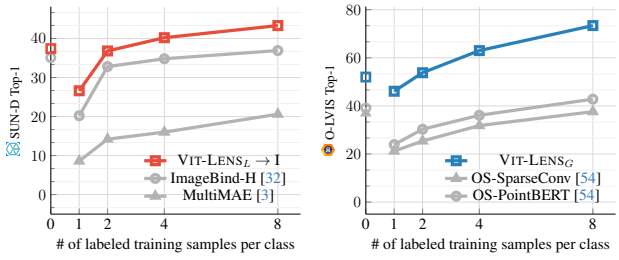


Figure 5. **Few-shot linear probing on depth and 3D point cloud.** We mark the zero-shot classification performance on the y-axis. We train linear classifiers on fixed features for the $\geq$1-shot settings.

We evaluate the label-efficiency of VIT-LENS through few-shot linear probing using VIT-LENS$_L$ (I) depth encoder and VIT-LENS$_G$ 3D encoder, as shown in Fig. 5. Linear classifiers are trained on fixed representation features. For few-shot depth linear probing, we compare with ImageBind [32] and MultiMAE [3] trained on images, depth, and semantic segmentation data. VIT-LENS consistently outperforms both methods in zero-shot and few-shot settings. For few-

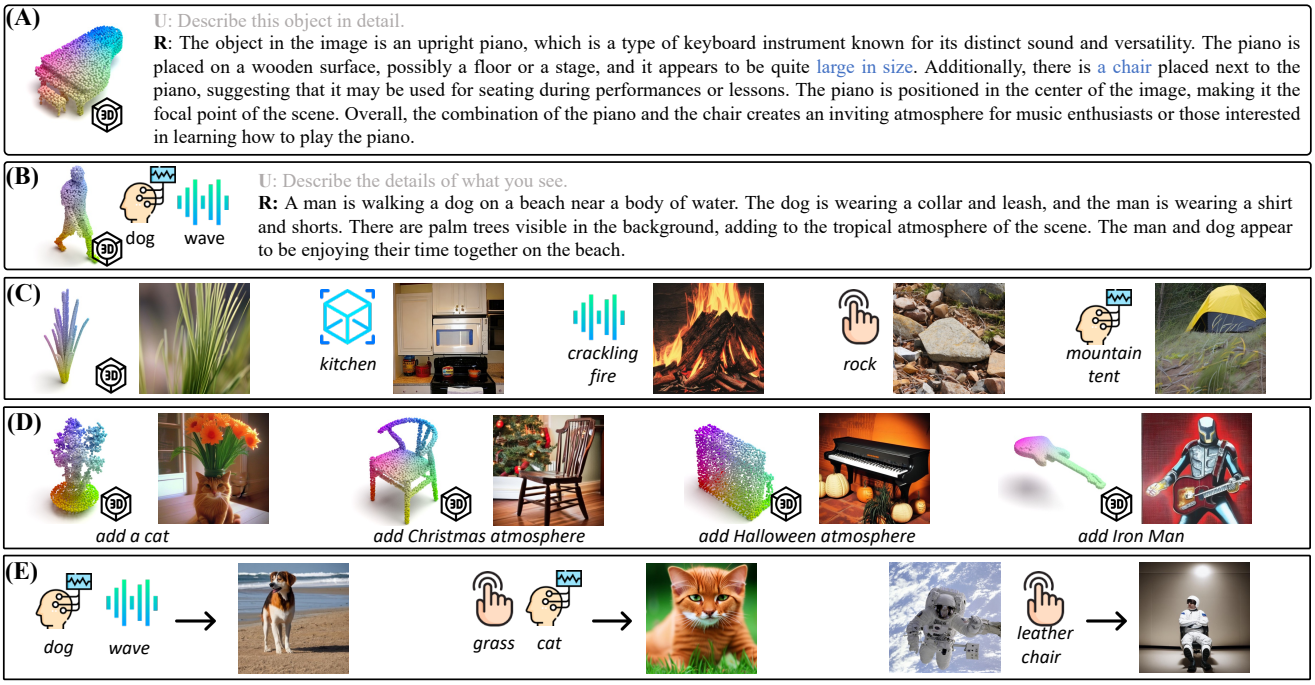


Figure 6. **Qualitative examples for plugging VIT-LENS into MFMs. (A-B) Integrate with InstructBLIP:** Accurately capturing concepts from single (A) or multiple modalities (B), providing detailed descriptions based on InstructBLIP's instruction-following capability. **(C-E) Integrate with SEED:** Extending SEED's capability to emergent compositional Any-to-image generation. (C) Single modality to image generation. (D) Text-guided any-to-image generation. (E) Multi-modalities-to-image generation.

shot 3D classification, we compare with SparseConv and PointBERT trained in [54]. VIT-LENS significantly outperforms all methods by a large margin in all few-shot settings, showcasing its robust generalization capabilities.

4.4. Results on VIT-LENS MFMs

In this section, we plug VIT-LENS across various modalities into off-the-shelf MFMs, and show in our experimental results that the MFMs' capabilities can be transferred to novel modalities and their combinations, without instruction-following training.

MFM Selection in Practice. In this work, we select InstructBLIP [16] and SEED [27, 28] to probe the emergent capabilities of the MFMs with our VIT-LENS plugged in. Both InstructBLIP and SEED utilize EVA01-g14 CLIP-ViT [24] as the visual encoder. Following the practice in Sec. 3.2, we use the same pretrained-ViT for VIT-LENS training in MFMs experiments. More details can be found in Supp.

InstructBLIP with VIT-LENS. InstructBLIP [16] introduced a framework for instruction tuning in a vision-language model, demonstrating its capabilities in tasks like complex visual reasoning and image descriptions. We show in our experiment that these capabilities can be effectively extended to novel modalities through the integration of VIT-LENS. Qualitative examples in Fig. 6 (A-B) showcase the model's ability to follow instructions across various modalities, enabling Any-modality QA, captioning, *etc*. Additionally, the model demonstrates precise and detailed descriptions, such as identifying a small "chair" next to a giant piano in (A), emphasizing the superior alignment achieved by VIT-LENS.

SEED with VIT-LENS. SEED-LLaMa [28] is an MFM distinguished by its capacity for multimodal comprehension and image generation. This is achieved through multimodal pretraining and instruction tuning along with its SEED tokenizer [27]. We present qualitative results of integrating VIT-LENS with SEED in Fig. 6 (E-G). The outcomes illustrate how the combined model extends SEED's capabilities to diverse modalities. Examples in (E-G) show the ability of compositional any-to-image generation [28]. It can translate input from any modality into an image, generate an image based on a text prompt given input from any modality, and seamlessly blend visual concepts from combinations of any modalities into a coherent and plausible image.

4.5. Ablation Study

We conduct ablations studies to investigate the effectiveness of various designs for VIT-LENS in omni-modal learning. We report the main results here and full details are in Supp.

Lens designs for different modalities. We study the effect of Lens designs as outlined in Sec. 3.1 for different modalities. We use VIT-LENS$_B$ and set comparable amount of

Test Dataset ▶	MN40	SUN-D	ESC	TAG-M	IN-EEG
S-Attn w/o pt weights	63.8	48.6	70.1	61.8	25.4
S-Attn w/ pt weights	65.4	50.9	70.9	63.6	26.3
Iter-CS-Attn	65.4	47.5	71.2	60.6	35.9

Table 8. **Lens designs** for different modalities. All modalities are aligned to "I+T". Lens w/ pt weights means tuning corresponding Self-Attn blocks in the pretrained-ViT, and w/o means random initialization. Default setting is marked with color box.

trainable parameters for the two variants. We also examine the effectiveness of initializing S-Attn type Lens with pretrained weights. We train 3D point cloud on ULIP-ShapeNet and follow the main settings for other modalities. The results are shown in Tab. 8. We observe consistent performance enhancement by initializing S-Attn Lens with pretrained weights. For image-like inputs such as depth maps and RGB-based tactile data, the S-Attn design exhibits superiority. Conversely, modalities significantly different from image inputs, like 3D point clouds, audio spectrograms, and EEG, benefit more from Iter-CS-Attn design. Additionally, it reduces the computational overhead by reducing the input length for ViT. Further details are available in the Supp.

Modality encoder designs and settings. We investigate the efficacy of integrating a set of pretrained-ViT layers into the modality encoder. We use the same datasets for training and testing as in the Lens design ablation. We compare VIT-LENS$_B$ with an architecture that combines ModEmbed and ViT and employing different settings for the ViT component, as detailed in Tab. 9. Results indicate that simply adding the ModEmbed to a pretrained-ViT cannot fully exploit the the potential of the pretrained-ViT (#2). Training the entire encoder with pretrained weights outperforms training from scratch, highlighting the effectiveness of utilizing the pretrained weights for learning (#1 vs #3). In comparison to #3, VIT-LENS$_B$ achieves comparable or better performance, especially for the less common modalities. Moreover, our VIT-LENS employs fewer trainable parameters than training the entire encoder and reduces computational overhead for modalities with lengthy inputs. Consequently, by introducing Lens, VIT-LENS effectively and efficiently transfers the capabilities of pretrained-ViT to various modalities.

Test Dataset ▶	MN40	SUN-D	ESC	TAG-M	IN-EEG
#1 M.E. → ViT (scratch)	62.4	46.3	68.8	55.6	20.5
#2 M.E. → ViT (pt, frozen)	50.0	36.8	54.9	24.8	14.2
#3 M.E. → ViT (pt, tune)	67.4	48.2	71.6	59.4	27.2
VIT-LENS$_B$	65.4	50.9	71.2	63.6	35.9

Table 9. **Encoder designs and settings** for different modalities. All modalities are aligned to "I+T". M.E. denotes ModEmbed. VIT-LENS$_B$ is the default setting.

Scaling up foundation model and VIT-LENS. We explore the effectiveness of scaling up VIT-LENS for feature alignment. We conduct experiments to pretrain for 3D on ULIP-ShapeNet, and depth on SUN-D. While previous works [32, 54] show that scaling to a large encoder(>100M) degrades the performance, we show in Fig. 7 that scaling up VIT-LENS can improve the 3D and depth representation and enhance performance.

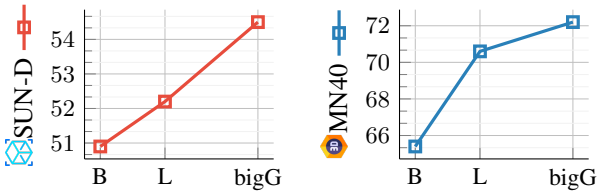


Figure 7. **Scaling the VIT-LENS on depth and 3D point cloud.** B: VIT-LENS$_B$, L: VIT-LENS$_L$, bigG: VIT-LENS$_G$.

Different pretrained-ViTs for VIT-LENS. We evaluate different pretrained-ViT variants for omni-modal representation learning. We use CLIP-ViT-bigG/14 as the teacher foundation model and apply different ViTs for the modalitiy encoder. We use the same datasets for training and testing as in the model scaling ablation. Results in Tab. 10 demonstrate that the use of pretrained-ViTs including the self-supervised and CLIP pretrained variants, outperforms training from scratch on both depth and 3D modalities. This indicates that different pretrained-ViTs possess the potential to serve as effective omni-modal learners.

ViT variant ▶	RndInit ViT-B16	DINO [7] ViT-B16	OpenCLIP ViT-B16	OpenCLIP ViT-L14	OpenCLIP ViT-bigG14
SUN-D	48.0	50.9	51.4	53.2	54.5
M40N	66.2	68.5	68.3	71.4	72.2

Table 10. **Different ViT** for modality encoders in VIT-LENS. we train the entire encoder for the baseline RndInit (random initialization) , while others follow VIT-LENS training setting.

5. Conclusion

In this paper, we introduce VIT-LENS, a straightforward yet effective method to advance omni-modal representations. VIT-LENS employs a pretrained-ViT to encode features for diverse modalities, eliminating the need for separate modality-specific architectures. The rich knowledge from large-scale data within the pretrained-ViT also reduces the burden of extensive data collection. We train VIT-LENS for various modalities, including 3D point cloud, depth, audio, tactile, and EEG. Experimental results in understanding tasks demonstrate that VIT-LENS consistently achieves leading performance. Moreover, we integrate VIT-LENS to off-the-shelf MFMs, *i.e.,* InstructBLIP and SEED, unlocking emergent capabilities such as any-modality instruction following, any-modality-to-image generation and text-guided any-modality-to-image editing. Finally, we believe that VIT-LENS will stimulate further research and innovation in the field of omni-modal representation learning, paving the way for more versatile and robust AI systems.

Appendix

A. More Details of VIT-LENS Method

A.1. ModEmbed for Modality Encoder

As describe in Sec. 3.1, we adopt a specific tokenization scheme to transform raw input signals into token embeddings for each modality. In this section, we introduce the modality embedding modules for 3D point cloud, depth, audio, tactile and EEG in our work.

3D point cloud. For 3D point cloud embedding, we utilize the approach introduced in [97]. We initially sample g center points from the input point cloud p using farthest point sampling (FPS). Subsequently, we utilize the k-nearest neighbors (kNN) algorithm to select k nearest neighbor points for each center point, forming g local patches $\{p_i\}_{i=1}^{g}$. To extract the structural patterns and spatial coordinates of these local patches, we normalize them by subtracting their center coordinates. Further, we employ a mini-PointNet [71] to project these sub-clouds into point embeddings. Additionally, we incorporate learnable positional embeddings on top of these embeddings, serving as inputs to standard Transformers or Lens models.

Audio. For audio embedding, following [34], we firstly convert the input audio waveform into a sequence of log Mel filterbank (fbank) features, forming a spectrogram with time and frequency dimensions. This spectrogram is then partitioned into a sequence of $P \times P$ patches with a stride of S in both time and frequency dimensions. Each $P \times P$ patch is flattened and projected into a 1D embedding of size d using a linear projection layer. Subsequently, we introduce learnable positional embeddings to capture the spatial structure of the spectrogram. These embeddings are utilized as inputs for subsequent processing by the model.

Depth. For depth embedding, we firstly follow [31, 32] to convert depth maps into disparity for scale normalization. We then utilize patch embedding similar to the mechanism in ViT. This involves partitioning the disparity into $P \times P$ patches with a stride of S ($S = P$) to handle the single-channel input. Each $P \times P$ patch undergoes flattening and projection into a 1D embedding of size d using a linear projection layer. To capture positional information, we incorporate learnable positional embeddings. These embeddings serve as inputs for the subsequent module.

Tactile. For tactile embedding, since we use RGB data from GelSight [43], we apply the same patch embedding as in ViT. Specifically, we partition the RGB input into $P \times P$ patches with a stride of S ($S = P$). Each $P \times P$ patch undergoes flattening and projection into a 1D embedding of size d using a linear projection layer. We integrate learnable positional embeddings for position information. These embeddings are forwarded as inputs for the subsequent module.

EEG. For EEG embedding, we use the C channel EEG with T timestamps. We then group every t time steps into a token and transformed it into a d-dimensional embedding. We further add positional embeddings on top and use the yielded embeddings as inputs for the subsequent module.

A.2. More Details for Lens

Reducing computational comprexity with Iter-CS-Attn. As is shown in Fig. 3 in the main paper, the cross attention mechanism generates an output with equal length of the latent query input. In practice, we typically configure the Lens with less parameters (fewer attention layers) compared to the pretrained-ViT component. Consequently, the majority of computational overhead is incurred during the forward pass of the ViT layers. Consider the input latent query length $\mathbf{n}$ and the modality embedding length $\mathbf{m}$. For modalities with lengthy input ($\mathbf{m} > \mathbf{n}$), utilizing the Iter-CS-Attn Lens reduces the computational cost of pretrained-ViT to $\mathcal{O}(\mathbf{n}^2)$, compared to encoding embeddings of the same length as the input, which has a complexity of $\mathcal{O}(\mathbf{m}^2)$. This strategy significantly lowers the computational overhead for processing lengthy inputs.

A.3. Utilizing Pretrained ViT Layers

The core of enhancing omni-modal representation with VIT-LENS is to leverage the rich knowledge encoded in the ViT that is pretrained on large-scale data. To integrate pretrained-ViT into the modality encoder, we apply the last l out of the total L transformer layers while maintaining a relatively high ratio $\frac{l}{L}$. This strategy draws inspiration from recent research exploring ViT interpretation [30, 77]. These studies revealed that ViT captures higher-level semantic concepts in its deeper layers while encoding general edges and textures in the shallower ones. Building upon these insights, we posit that the shared high-level knowledge among different modalities is mostly preserved in the deeper layers of the ViT architecture. Consequently, we propose the utilization of a set of pretrained-ViT layers within the modality encoder in our pipeline. Notably, when $\frac{l}{L} < 1$, we either discard the initial $L - l$ transformer layers or integrate them for S-Attn type Lens learning if applicable.

B. More Experimental Details and Results

B.1. Datasets and Metrics

ULIP-ShapeNet Triplets [92]. The ULIP-ShapeNet Triplets training data for 3D point cloud is derived from ShapeNet55 [8] by Xue *et al*. [92]. All the 3D point clouds are generated from CAD models. Anchor images are synthesized using virtual cameras positioned around each object, and texts are obtained by filling metadata into a predefined prompt template. This dataset comprises approximately 52.5k 3D point cloud instances.

ULIP2-Objaberse Triplets [93]. The ULIP2-Objaverse Triplets training data for 3D point cloud is developed by Xue *et al*. [93], utilizing the recently released Objaverse [17]. For each 3D object, 12 rendered images are obtained, spaced equally by 360/12 degrees. Each rendered image has 10 detailed captions generated using BLIP2-opt6.7B [48]. It includes around 798.8k 3D point cloud instances.

OpenShape Triplets [54]. The OpenShape Triplets training data for 3D point clouds encompasses four prominent public 3D datasets: ShapeNet [8], 3D-FUTURE [25], ABO [14] and Objaverse [17]. For each 3D object, 12 color images are rendered from preset camera poses, and thumbnail images are included as candidates if provided. OpenShape employs various strategies to obtain high-quality text descriptions, including filtering noisy metadata using GPT4 [64], generating captions using BLIP [47] and Azure cognition services, and conducting image retrieval on LAION-5B to retrieve relevant texts with paired images closely resembling the object's rendered image, leading to a wider range of text styles. This dataset comprises approximately 876k 3D point cloud instances.

ModelNet40 [90]. The ModelNet40 dataset is a widely used benchmark in the field of 3D object recognition. It consists of 12,311 CAD models from 40 categories, with 9,843 training samples and 2,468 testing samples. It includes everyday objects such as chairs, tables, desks, and other household items. Each object is represented as a 3D point cloud and has been manually annotated with the object's category. The dataset is commonly used for tasks like shape classification and shape retrieval. In this work, we only use the test samples for zero-shot classification. The evaluation is performed using Top-K accuracy.

ScanObjectNN [85]. The ScanObjectNN dataset is a significant resource in the domain of 3D object recognition and segmentation. It encompasses a diverse array of 3D object instances acquired through a commodity RGB-D camera. This dataset exhibits a wide spectrum of household items, furniture, and common indoor objects. Each individual object instance is annotated with fine-grained semantic and instance-level labels. In total, it contains 2,902 objects distributed across 15 distinct categories. In this work, we follow [54] to use the variant provided by [97] for zero-shot classification, which contains 581 test shapes with 15 categories. The evaluation is performed using Top-K accuracy.

Objaverse-LVIS [17]. This dataset is an annotated subset of Objaverse [17] and consists of 46,832 shapes from 1,156 LVIS [37] categories. With a larger base of classes compared to other benchmarks, Objaverse-LVIS presents a challenging long-tailed distribution, making it a better reflection of the model's performance in open-world scenarios. In this work, we follow [54] to use this dataset for zero-shot classification, and the evaluation is performed using Top-K accuracy.

SUN-RGBD [78]. We utilize paired RGB and depth maps along with associated class labels from the SUN-RGBD dataset. For training VIT-LENS, we employ the train set comprising approximately 5k samples. To evaluate classification performance, we use the test set (**SUN Depth-only**), which contains 4,660 samples. Specifically for testing, we only utilize depth as input and construct classification templates using the 19 scene classes available in the dataset. The evaluation process involves Top-K accuracy metrics.

NYU-Depth v2 [61]. We utilize the depth maps from NYU-Depth v2 test set (**NYU-v2 Depth-only**) containing 654 samples for evaluation. We use 16 semantic classes in the dataset and follow previous work [32] to conduct 10-class classification. Concretely, for classification, there is an "others" class corresponding to 7 different semantic classes – ['computer room', 'study', 'playroom', 'office kitchen', 'reception room', 'lobby', 'study space']. For classification, we compute the similarity of the "others" class as the maximum cosine similarity among these 7 class names. We report Top-K accuracy.

Audioset [29]. This dataset is utilized for both training and evaluation in our work. It contains 10-second videos sourced from YouTube and is annotated across 527 classes. It consists of 3 pre-defined splits – unbalanced-train split with about 2M videos, balanced-train with about 20k videos and test split with about 18k videos. Due to the unavailability of some videos for download, we finally have 1.69M/18.7k/17.1k for these three splits. We use the train splits for training and the test split for evaluation. During evaluation and when textual data serves as anchor data during training, we make use of the textual class names along with templates. The evaluation metric employed is mean Average Precision (mAP).

ESC 5-folds [29]. The ESC50 dataset is a widely used benchmark dataset in the field of environmental sound classification. It comprises a collection of 2,000 sound recordings, categorically organized into 50 classes, including animal vocalizations, natural soundscapes, and human-made sounds. Each class in the dataset contains 40 audio samples that are five seconds long. It has pre-defined 5 fold evaluation, each consisting of 400 test audio clips. In this work, we evaluate the zero-shot prediction on across the 5 folds and report the overall Top-1 accuracy.

Clotho [22]. The Clotho dataset is an audio collection paired with rich textual descriptions, comprising a development set of 2,893 audio clips and a test set of 1,045 audio clips. Each audio clip is associated with five descriptions. In this study, we focus on the text-to-audio retrieval task. For evaluation, we treat each of the five associated captions as an individual test query, searching within the set of audio clips. We employ recall@K as the evaluation metric, where a query is considered successful if the ground truth audio is retrieved among the top-K returned audio clips.

AudipCaps [44]. This dataset comprises audio-visual clips sourced from YouTube, accompanied by textual descriptions. It features clips extracted from the Audioset dataset. In this study, we employed the dataset splits outlined in [62], specifically excluding clips that intersected with the VGGSound dataset. We end up with 813 clips in the test split for zero-shot evaluation. The task is text-to-audio retrieval and is evaluated by the recall@K metric.

VGGSound [9]. This is an audio-visual dataset sourced from YouYube. It contains more around 200k video clips of 10s long. These clips are annotated into 309 classes, covering a spectrum from human actions to sound-emitting objects and human-object interactions. Since some videos are no long available for downloading, we finally end up with 162k clips for train set and 15.5k for test set. In this work, the audio from the test set is utilized specifically for zero-shot classification tasks, evaluating model performance using the Top-1 accuracy metric.

Touch-and-go [94]. The Touch-and-Go dataset comprises real-world visual and tactile data gathered by human data collectors probing objects in natural settings using tactile sensors while simultaneously recording egocentric video. It offers annotations for 20 material classes, and provide hard/soft (H/S) and rougH/Smooth (R/S) labels. The dataset is organized into distinct splits: train-material and train-H/S with 92k samples, test-material and test-H/S with 30k samples, train-R/S with 35k samples and test-R/S with 8k samples. In our work, we utilize the train-material split for training and perform classification on the test-material subset. For zero-shot classification, we employ test-H/S and test-R/S subsets. In the context of linear probing, we fine-tune the model using the corresponding train set for a particular task. Our evaluation of model performance utilizes the Top-1 accuracy metric.

ImageNet-EEG [79]. This dataset comprises EEG recordings obtained from six subjects while they were presented with 2,000 images across 40 categories from the ImageNet dataset [18]. Each category contains 50 distinct images, resulting in a total of 12,000 128-channel EEG sequences. Recorded using a 128-channel Brainvision EEG system, the dataset covers diverse object categories, including animals (such as dogs, cats, elephants), vehicles (including airliners, bikes, cars), and everyday objects (such as computers, chairs, mugs). We leverage the observed image and/or its corresponding text label as anchor data. We conduct classification tasks on both the validation set (consisting of 1,998 samples) and the test set (consisting of 1,997 samples). Our evaluation of model performance is based on the Top-1 accuracy metric.

B.2. Data Input and Augmentation

Image and Video. When handling modalities such as images, videos, or tactile sensor data with RGB or RGBT inputs, we adopt the standard input representation used in the vanilla ViT model. Specifically, for image input, we partition it into patches of size $P \times P$. For video input, we employ 2-frame clips similar to the approach outlined in [32]. We construct patches of size $T \times P \times P$. Notably, $T = 2$, $P = 16$ for VIT-LENS$_B$, and $P = 14$ for VIT-LENS$_L$ and VIT-LENS$_G$. We inflate the visual encoder's weights to to accommodate spatiotemporal patches for video inputs. During inference, we aggregate features over multiple 2-frame clips. This adaptation enables models initially trained on image-text data to effectively handle videos.

3D point cloud. For 3D point cloud input, we follow previous work to uniformly sample 8,192 points [92, 93] or 10,000 points [54] as the input for 3D shape. During training, we apply standard augmentation [92] for the point clouds. As mentioned in Sec. A.1, we construct local patches by sampling 512 sub-clouds, each comprising 32 points. This is accomplished by employing Farthest Point Sampling (FPS) and the k-Nearest Neighbors (kNN) algorithm.

Depth. For the single-view depth, we follow [32] to use the in-filled depth and convert them into disparity. During training, when image is used as anchor data, we apply strong data augmentation for the anchor image, including RandAug [15] and RandErase [103]. We used aligned spatial crop for image and depth. For embedding module, we follow the CLIP-ViT to set $P = 16$ for VIT-LENS$_B$ and $P = 14$ for VIT-LENS$_L$.

Audio. For audio input, we process each raw audio waveform by sampling it at 16kHZ, followed by extracting a log mel spectrogram with 128 frequency bins using a 25ms Hamming window with the a hop length of 10ms. Consequently, for an audio duration of t seconds, our input dimensionality becomes $128 \times 100t$. During training, we randomly sample a 5-second clip for audio input, and apply spectrogram masking [66] with max time mask length of 48 frames and max frequency mask length of 12 bins. When image is used as anchor data, we randomly sample 1 frame from the corresponding clip and apply RandAug [15] for the sampled frame. We also apply Mixup [99] during training for both audio and its anchor data, with a mixup ratio of 0.5. For embedding module, we set $P = 16$ for VIT-LENS$_B$ and $P = 14$ for VIT-LENS$_L$, and $S = 10$. At inference time, we uniformly sample multiple clips to cover the full length of the input sample and aggregate the features extracted from these clips.

Tactile. For data from tactile sensors, we treat it similarly to RGB images. During training, we introduce random flips along the horizontal and vertical directions to augment the tactile input. Additionally, random rotations are applied to further augment the input data. When image is used as anchor data for training, we apply RandAug [15] to augment the image. For embedding module, we follow the CLIP-ViT to set $P = 16$ for VIT-LENS$_B$ and $P = 14$ for VIT-LENS$_L$.

EEG. For EEG input data, we follow [4] to use the 128-channel EEG sequences. These EEG signals are filtered

	3D Point Cloud	*Depth*	*Audio*	*Tactile*	*EEG*
ModEmbed ▶	Mini PointNet	PatchEmbed	PatchEmbed	PatchEmbed	Conv1D
Lens Config ▶	Iter-CS-Attn $N = 4, M = 1$ ✓tie weights	S-Attn $N = 4$ CLIP-ViT `Block.1-4` Init	Iter-CS-Attn $N = 2, M = 3$ -	S-Attn $N = 4$ CLIP-ViT `Block.1-4` Init	Iter-CS-Attn $N = 1, M = 1$ -
Pretrained ViT Config ▶	CLIP-ViT `Block.1-12`	CLIP-ViT `Block.5-12`	CLIP-ViT `Block.1-12`	CLIP-ViT `Block.5-12`	CLIP-ViT `Block.1-12`
VIT-LENS$_B$					
# Trainable Param.	34.1M	28.7M	72.1M	29.1M	17.4M
# Total Param.	119.7M	85.9M	157.7M	86.2M	103.0M
Flops	75.4G	36.5G	64.7G	36.6G	41.1G
VIT-LENS$_L$					
# Trainable Param.	60.0M	50.9M	127.6M	51.3M	30.6M
# Total Param.	363.4M	303.8M	431.0M	304.0M	333.9M
Flops	236.7G	168.6G	233.6G	168.8G	183.7G

Table 11. **Model Configuration for VIT-LENS.** We show the model configurations for the modality encoder across 3D point cloud, depth, audio, tactile, and EEG, for both VIT-LENS$_B$ and VIT-LENS$_L$ architectures. For modality embedding module, we list the name of architecture. For modality Lens configuration, we specify the adopted Lens type. For S-Attn type, N denotes the number of self-attention layers, accompanied by details on weight initialization. For Iter-CS-Attn type, N represents the number of basis blocks and M denotes the number of self-attention layers within each basis block. The term "tie weights" means parameter sharing among blocks ≥ 2 [41]. For the pretrained-ViT configuration, we showcase the set of frozen transformer layers used in the modality encoder. With the listed configurations, we show the number of trainable parameters, the number of total parameters and Flops for each modality encoder.

	3D Point Cloud	*Depth*	*Audio*	*Tactile*	*EEG*
Optimizer	AdamW				
Optimizer momentum	$\beta_1 = 0.9, \beta_2 = 0.98$				
Peak LR	5e-4/5e-4/2e-4*	2e-4	2e-4	2e-4	2e-4
Weight decay	0.2◇				
Batch size	512	512	2048	512	512
Warmup steps	10,000				
Sample replication	1	50	1	50	50
Total epochs	200/150/150*	100	80	40	40
Modality augmentation	RandDropout RandScale RandShift * RandPerturb RandRotate	RandResizeCrop(size=224) RandHorizontalFlip(p=0.5) RandAugment(m=9, n=2) RandErasing(p=0.25)	Frequency masking(12) Time masking(48) NoiseAug mixup(p=0.5)	RandResizeCrop(size=224) RandHorizontalFlip(p=0.5) RandVerticalFlip(p=0.5) RandRotation(degrees=(0,360))	-
Image augmentation	RandResizeCrop(size=224)*	RandResizeCrop(size=224) RandHorizontalFlip(p=0.5) RandAugment(m=9,n=2) ColorJitter(0.4) RandErasing(p=0.25)	RandShortSideScale(min=256, max=340) RandCrop(size=224) RandHorizontalFlip(p=0.5) RandAugment(m=9, n=2, p=0.3) mixup(p=0.5)	RandResizeCrop(size=224) RandHorizontalFlip(p=0.5) RandAugment(m=9,n=2) ColorJitter(0.4)	RandResizeCrop(size=224) RandHorizontalFlip(p=0.5) RandAugment(m=9,n=2) ColorJitter(0.4)

Table 12. **Training hyper-parameters for each modality.** ⋆ Separate hyper-parameters are reported for 3D training with different datasets: ULIP-ShapeNet, ULIP2-Objaverse, and OpenShape Triplets. * Augmentations listed for 3D training are applied to ULIP-ShapeNet and ULIP2-Objaverse, while released features are used for training on OpenShape Triplets. ◇ Weight decay excludes parameters for BatchNorm, LayerNorm, bias terms, and logit scale.

within the frequency range of 5-95Hz and truncated into a common length of 512. For embedding module,we utilize Conv1D, configuring the kernel size to 1 and the stride to 1.

B.3. Model Configuration

In this section, we provide the configurations for encoders of different modalities in VIT-LENS. Details are specified in Tab. 11.

B.4. Training Setup

In Tab. 12, we list the hyper-parameters used in training for each modality. Our experiments were done on 32GB V100 GPU clusters.

B.5. Additional Results on Understanding Tasks

Additional results on audio tasks. We merged Audioset [29] and VGGSound [9], resulting in a combined training dataset with 1.86M samples. Results in Tab. 13 show improved performance across all benchmarks due to the inclusion of additional training data.

Training data	AS-A	VGGS	ESC	Clotho		AudioCaps	
	mAP	Top1	Top1	R@1	R@10	R@1	R@10
AS	26.7	31.7	75.9	8.1	31.2	14.4	54.9
AS + VGGS	27.2	51.7	80.9	7.9	31.5	14.9	55.2

Table 13. **Training for audio with more data.** We Audioset(AS) and Audioset+VGGSound(AS+VGGS) to train VIT-LENS$_L$ and report metrics following Tab. 3.

B.6. More Details and Results for VIT-LENS MFMs

Architectural details for VIT-LENS MFM integration. Both InstructBLIP [16] and SEED [27, 28] apply the pretrained EVA01-g14 [24] CLIP-ViT to perceive and encode images for the subsequent LLM input. Concretely, they use the first 39 transformer layers of the 40-layer CLIP-ViT for visual feature extraction. Adhering to this configuration, we employ the EVA01-g14 CLIP as the foundation model and utilize its CLIP-ViT as an integral part of the modality encoder for the training of multimodal alignment. We tune the parameters of ModEmbed and Lens. During inference, we directly plug the ModEmbed and Lens prior to the pretrained-ViT, enabling the yielded MFM to handle inputs of various modality without specific instruction following.

B.6.1 Additional Results: InstructBLIP with VIT-LENS

Comparison of InstructBLIP with VIT-LENS against other methods on 3D data instruction following. We train VIT-LENS for 3D point cloud using ULIP2-Objaverse and integrate it into InstructBLIP. Beyond capturing the high-level semantics of the input data, we observed that leveraging the EVA01-g14 CLIP-ViT within the modality encoder further enhanced the model's ability to capture local details.

Our qualitative evaluation involves a comparison with: (1) PointBERT [97] aligned with EVA01-g14 CLIP, replacing the vision encoder used in InstructBLIP; and (2) CLIP-Cap [58] from OpenShape [54]. We present a snapshot of qualitative outcomes across different models in Tab. 14, Tab. 15 and Tab. 16. These examples showcase several capabilities exhibited by VIT-LENS integration without specific tuning using 3D-related instructional data. Notably, the examples demonstrate that VIT-LENS empowers InstructBLIP to accurately describe 3D objects. For instance, the plant example in Tab. 15 is characterized as "sitting in a ceramic pot" and "bamboo-like". Moreover, VIT-LENS excels in capturing local visual concepts beyond the most prominent ones. For instance, the piano example in Tab. 14 describes the observation of a "chair".

For PointBERT integrated InstructBLIP, although PointBERT achieves decent performance for zero-shot classification, it fails to provide accurate information for the InstructBLIP as VIT-LENS does. We can see that in Tab. 14, although it recognizes the piano, it fails to provide accurate brief and detailed description since it includes "person" in its description, which does not exist in the 3D input. Also, it fails to recognize the plant in Tab. 15 and the toilet in Tab. 16.

CLIPCap-OpenShape, while occasionally displaying some relevant entities in captions ("vase" in Tab. 15 and "toilet" in Tab. 16), often generates hallucinations and inaccurate captions.

The overall results demonstrate that VIT-LENS excels not only at classifying the salient object of the 3D input, but also capturing the visual details. This merit is surprising: despite the fact that we only explicitly use the `[CLS]` for alignment, the integrated model exhibits the ability to capturing local information. This ability might stem from VIT-LENS potentially inheriting information captured by other tokens, which could carry local details to the input of InstructBLIP. This capability indicates that the model might leverage the collective knowledge present in various tokens, not limited to the `[CLS]`, contributing to its robustness in encoding rich visual information.

InstructBLIP with VIT-LENS for input of multiple modalities. We demonstrate that the versatile omni-modal VIT-LENS encoder, coupled with an array of specialized Lenses, functions as a sensor adept at concurrently perceiving and understanding multiple modalities. To achieve this, we concatenate the outputs from diverse modality Lenses prior to inputting them into the ViT transformer. Subsequently, the encoded embeddings from this concatenation are forwarded to the LLM within InstructBLIP for text generation.

Qualitative results[2] are showcased in Tab. 17 for dual-modality input and in Tab. 18 for tri-modal input. The outputs produced by InstructBLIP with VIT-LENS underscore its remarkable ability to concurrently interpret multiple modalities, akin to perceiving an image. Notably, as evident in the qualitative results, the incorporation of VIT-LENS enhances the resulting MFM's capacity to digest multi-modal inputs, discover unconventional co-occurrence of concepts from different modalities, and craft stories based on the aggregated information from multiple modalities without specific instruction tuning.

B.6.2 Additional Results: SEED with VIT-LENS

Integrating the well-trained modality Lenses into SEED enables seamless any-modality to image generation. In addition to the primary paper, we present further qualitative results in Fig. 8.

In (S-A), we showcase the capability of generating images from any modality input by encoding the modality information using Lens and the SEED tokenizer [27]. We append to this with a text prompt, "`Generate an image based on what you perceive.`" The results demonstrate

[2]Photos credited to https://www.pexels.com/.

that our VIT-LENS integrated MFM successfully generates images across diverse modalities, including 3D point clouds, audio, EEG, tactile, and depth. Notably, in the 3D point cloud examples (Row 1 in S-A), the model retains the local structure of the 3D shapes. Furthermore, the model exhibits the ability to generate diverse images for different inputs within the same category, exemplified by the audio examples for "sea waves", "engine", "crackling fire" and more. This showcases the robustness of our VIT-LENS.

In (S-B), we show that our integration extends SEED's capability for compositional image generation to any modality. For better visualization, we show examples for 3D point clouds. We show the main instruction under each example. In practice, we feed any modality input into the LLM via Lens and the SEED tokenizer, supplementing it with the prompt "`[Instruction], generate an image`" to guide text-based generation. The presented results highlight the model's ability to retain visual concepts accurately in the generated images. Additionally, it preserves the local structure rather than merely focusing on high-level semantics. Notably, in examples such as "guitar" and "car" the model successfully retains the shape and some local structures, showcasing its nuanced understanding beyond high-level semantic understanding.

Moreover, the model demonstrates the capability to intake inputs from various modalities and subsequently generate an image that combines all the conveyed concepts in a coherent manner. In practice, we employ the prompt "`[input tokens A], [input tokens B], please generate an image to combine them`" to facilitate this process. For a visual examples, please refer to Fig. 6-(E) in the main paper.

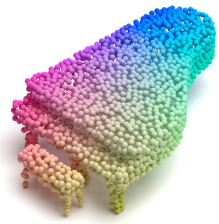
Visual input example, Piano:	
	
User	What is this object?
InstructBLIP w/ VIT-LENS	Piano.
User	What is the function of this?
InstructBLIP w/ VIT-LENS	Pianos are used to play music.
User	Describe it in detail.
InstructBLIP w/ VIT-LENS	The object in the image is an upright piano, which is a type of keyboard instrument known for its distinct sound and versatility. The piano is placed on a wooden surface, possibly a floor or a stage, and it appears to be quite large in size. Additionally, there is a chair placed next to the piano, suggesting that it may be used for seating during performances or lessons. The piano is positioned in the center of the image, making it the focal point of the scene. Overall, the combination of the piano and the chair creates an inviting atmosphere for music enthusiasts or those interested in learning how to play the piano.
User	What is this object?
InstructBLIP w/ PointBERT	Advertisement.
User	Describe this object briefly.
InstructBLIP w/ PointBERT	A man sitting at a desk with a piano in the background.
User	Describe this object in detail.
InstructBLIP w/ PointBERT	This scene showcases a person sitting at a desk with a piano in the background. The person is wearing a suit and tie, indicating that they may be attending an important business meeting or event. The scene is set in a professional environment, emphasizing the importance of proper attire and etiquette. Additionally, there is a laptop on the desk, suggesting that the person may be working on various tasks related to their job or responsibilities.
CLIPCap-Openshape [54, 58]	A white house with a microphone in the middle of a dark room.

Table 14. Example for VIT-LENS enabling InstructBLIP to understand 3D inputs.

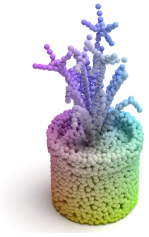
Visual input example, Plant:	
	
User	What is this object?
InstructBLIP w/ VIT-LENS	Potted plant.
User	Describe it in detail.
InstructBLIP w/ VIT-LENS	In the image, there is a potted plant sitting in a ceramic pot on a table. The plant appears to be a bamboo-like species, and it is positioned in such a way that its leaves appear to be growing out of the top of the pot. The plant's placement on the table suggests that it is meant to be a decorative element, adding a touch of greenery to the space. The ceramic pot has a diameter of around 10 centimeters, making it a suitable size for holding a small plant like this one. Overall, the image showcases a simple yet elegant arrangement of a potted plant in a ceramic pot on a table.
User	What is this object?
InstructBLIP w/ PointBERT	1000px by 1000px
User	Describe this object briefly.
InstructBLIP w/ PointBERT	A man in a suit and tie is standing in front of a desk.
CLIPCap-Openshape [54, 58]	Flowers in a vase, black and white.

Table 15. Example for VIT-LENS enabling InstructBLIP to understand 3D inputs.

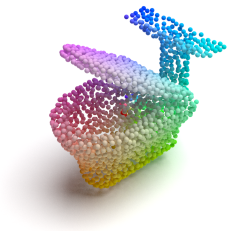
Visual input example, Toilet:	
	
User	What is in the scene?
InstructBLIP w/ VIT-LENS	Toilet.
User	Describe it in detail.
InstructBLIP w/ VIT-LENS	The image features a close-up view of a toilet with its lid open, revealing the inside of the toilet bowl. The toilet appears to be made of porcelain or ceramic material, and it has a seat attached to it. The toilet is positioned on a white background, providing a clear and detailed view of the toilet's design and construction. In addition to the toilet, there is another object in the image, which could be a piece of furniture or a decorative item, but it is not as prominent as the toilet. Overall, the image showcases a functional and well-designed toilet, emphasizing the importance of cleanliness and hygiene in daily life.
User	What is in the scene?
InstructBLIP w/ PointBERT	Lamp.
User	Describe this object briefly.
InstructBLIP w/ PointBERT	An image of a desk with a lamp on it.
CLIPCap-Openshape [54, 58]	The moment a man's hand reaches out to touch a toilet bowl.

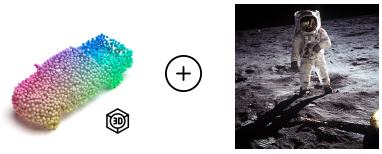
Table 16. Example for VIT-LENS enabling InstructBLIP to understand 3D inputs.

InstructBLIP w/ VIT-LENS, Two Multimodal Inputs.

Example 1: 3D Point Cloud + Image

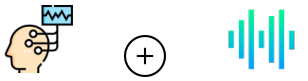
Prompt 1: Describe based on what you see.
Response 1: An image of a piano with the earth in the background.
Prompt 2: What is unusual about this image?
Response 2: The unusual aspect of this image is that it depicts a piano floating in space, with the Earth visible in the background. This is an imaginative and creative representation of a musical instrument in an unconventional setting, showcasing its versatility and adaptability to different environments.
Prompt 3: Imagine the events leading up to this moment and those that follow. Create a story that links them together.
Response 3: In the image, we see a grand piano placed on top of the earth, with the moon and stars visible in the background. This setting creates a unique and captivating scene that invites the viewer to imagine the events leading up to this moment and those that follow. One possible storyline could involve a talented pianist who has always dreamed of performing on top of the earth. After years of hard work and dedication, they finally have the opportunity to make their dream come true. As they prepare for their performance, they reflect on the journey that led them to this moment and the challenges they faced along the way. During their performance, they play a beautiful melody that captures the essence of the earth's beauty and the vastness of the universe. The audience is mesmerized by the performance, and the pianist feels a sense of accomplishment and fulfillment as they conclude their performance on top of the earth.

Example 2: 3D Point Cloud + Image



Prompt 1: Imagine you are a narrator looking at this, describe the essence of what you see.
Response 1: An astronaut standing next to a pickup truck on the surface of the moon.
Prompt 2: What is unusual aboout this image?
Response 2: The unusual aspect of this image is that it depicts an astronaut standing on the surface of the moon with a pickup truck in the foreground. This combination of elements is not commonly associated with space exploration or the moon's surface, making it an unconventional and eye-catching image.
Prompt 3: Craft a narrative based on the visual elements in the picture. Let your imagination guide the story.
Response 3: In the image, an astronaut is standing on the moon's surface with a pickup truck in the background. The astronaut is wearing a spacesuit and appears to be exploring the lunar terrain. The pickup truck is parked on the moon's surface, adding a unique touch to the scene. The combination of the astronaut and the pickup truck creates an intriguing narrative that invites the viewer to imagine what might be happening in this futuristic setting.

Example 3: EEG + Audio

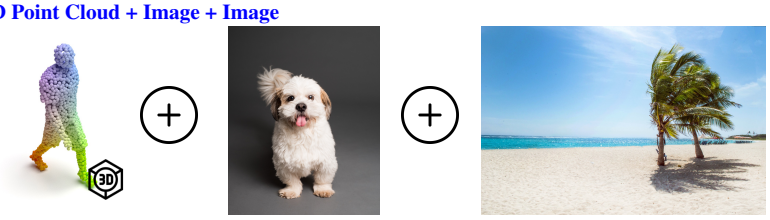


Prompt 1: Tell me based on what you see.
Response 1: A small dog sitting on top of a sandy beach.
Prompt 2: Tell me more about this scene.
Response 2: In the image, we see a cute little dog sitting on a sandy beach. The dog is wearing a blue collar and appears to be waiting patiently for its owner or someone else to come along. The beach is littered with small pebbles, adding to the serene atmosphere of the scene. As the sun sets in the background, the sky takes on a beautiful orange and pink hue, casting a warm and inviting glow over the entire area. The dog seems to be enjoying the peaceful surroundings, taking in the sights and sounds of the beach. It's a heartwarming image that captures the beauty of nature and the joy of spending time with a beloved pet.

Table 17. Example to illustrate the Instruct-BLIP with input of two modalities.

InstructBLIP w/ VIT-LENS, Three Multimodal Inputs.

Example 1: 3D Point Cloud + Image + Image



Prompt 1: Describe the scene based on what you see.
Response 1: A man walking a dog on a beach with palm trees in the background.
Prompt 2: Describe the details of what you see.
Response 2: The image depicts a man and his dog walking on a sandy beach. The man is holding the dog's leash, while the dog appears to be enjoying its time on the beach. In the background, there is a palm tree visible, adding to the serene atmosphere of the scene. The man and the dog seem to be in good spirits, enjoying each other's company as they stroll along the beach.
Prompt 3: Imagine the events leading up to this moment and those that follow. Create a story that links them together.
Response 3: In the story, the man and his dog are walking along the beach, enjoying the sunny day. As they walk, the man notices a piece of paper lying on the sand. Curious, he picks it up and discovers that it's an advertisement for a photography contest. The man decides to enter the contest and submits a photo of himself and his dog on the beach. To his surprise, his photo is selected as one of the winners, and he is invited to attend the award ceremony. At the event, he meets other photographers and learns more about the art of photography. Through this experience, the man gains a new appreciation for photography and decides to pursue it as a hobby. He continues to take photos of his dog and other subjects, honing his skills and capturing unique moments in time. Eventually, his passion for photography leads him to become a professional photographer, and he shares his work with others, inspiring them to appreciate the beauty of the world around them.

Example 2: EEG + Tactile + 3D Point Cloud

Prompt 1: Describe what you see.
Response 1: A person and a dog walking in a grassy area.
Prompt 2: Create a short story around the given visual contents.
Response 2: This scene depicts a man walking his dog in a grassy field. The man and the dog are both visible in the scene, with the man standing on the left side of the image and the dog standing on the right side. The man is wearing a shirt and pants, while the dog is wearing a collar. The man and the dog seem to be enjoying each other's company as they walk together in the field.

Example 3: EEG + Audio + Audio

Prompt 1: Describe what you see.
Response 1: A white horse running on the beach with fireworks in the background.
Prompt 2: Create a short story around the given visual contents.
Response 2: The image depicts a white horse standing on a beach, surrounded by water and trees. The horse appears to be in motion, possibly running or galloping across the beach. In the background, there is a firework display taking place, adding to the festive atmosphere of the scene. The combination of the horse, the beach, and the fireworks creates a unique and visually captivating image.

Table 18. Example to illustrate the Instruct-BLIP with input of three modalities.

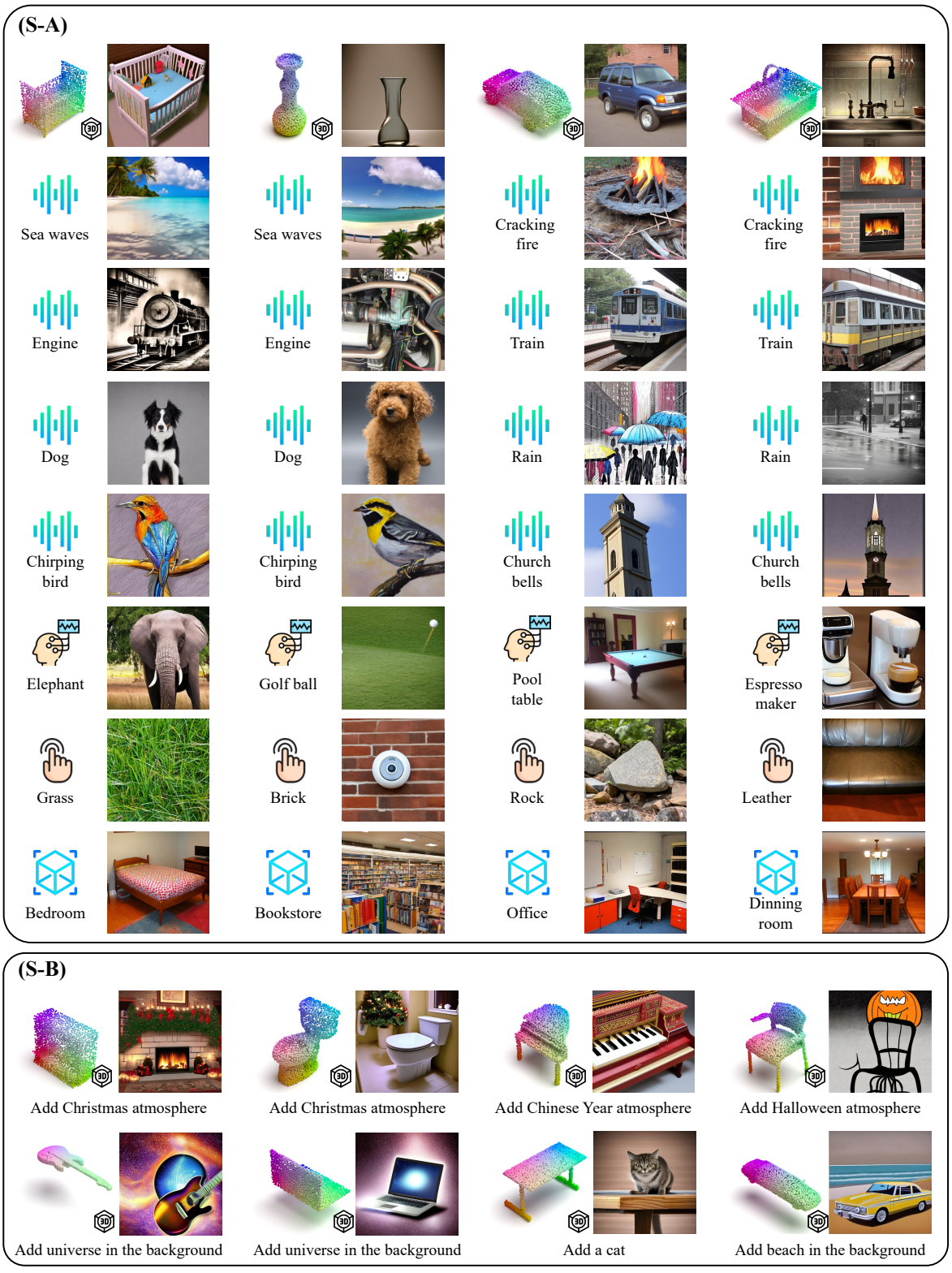


Figure 8. **Qualitative examples for plugging VIT-LENS into SEED.** We present the input-output pairs in a local left-right pattern. **(S-A) Any modality to image generation.** The integrated model generates an image output (right) corresponding to the provided individual input (left). **(S-B) Compositional any modality to image generation.** We focus on 3D point cloud cases in the examples for better visualization. The integrated model generates a corresponding image (right) when presented with the input (left) along with the conditioned text prompt.

B.7. Applications

The versatility of VIT-LENS in binding diverse modalities into a unified space unlocks a multitude of applications, including cross-modal retrieval and semantic search. This section demonstrates the application of VIT-LENS in the domain of any-modality to 3D scene understanding, leveraging the capabilities of the recent OpenScene framework [68]. OpenScene aligns 3D point features within the CLIP embedding space, enabling text-based and image-based searches within a 3D scene. Building upon OpenScene, VIT-LENS extends this understanding of 3D scenes to encompass more modalities.

The qualitative results in Fig. 9 demonstrate this application's ability to utilize inputs from multiple modalities to identify relevant areas within the scene. It effectively highlights objects like the toilet flush based on toilet audio, the sink area using 3D point cloud data of a water sink, the kitchen area from the depth map, and the presence of sofas inferred from tactile input indicating a leather sofa.

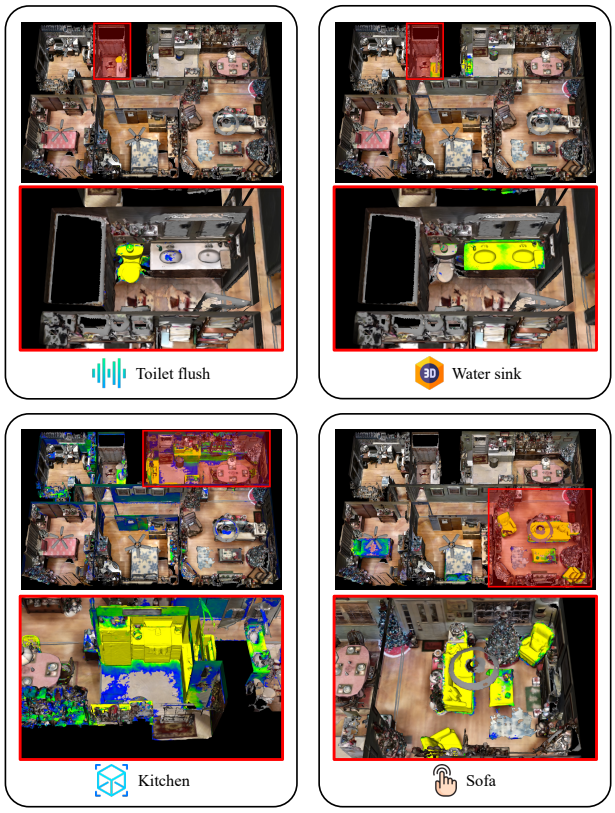


Figure 9. **Application for any-modality to 3D scene understanding.** This application facilitates scene exploration by accepting inputs from diverse modalities and subsequently highlighting relevant areas within the scene. In the visualization, the color gradient represents the relevance level within the scene (yellow is the highest, green is moderate, blue is low, and uncolored is lowest).

B.8. Additional Ablation Studies

This section presents additional ablation experiment findings regarding VIT-LENS training.

B.8.1 Anchor Data for Alignment

We study the effect of using different anchor data for multi-modal alignment during training. We employ VIT-LENS$_B$ in experiments. We train for 3D point cloud on ULIP-ShapeNet and follow the main settings for other modalities. The results are shown in Tab. 19. Our observations reveal that employing both image and text as anchor data yields superior performance for tasks involving 3D point clouds, depth, and audio. In contrast, utilizing only image or text alone results in comparatively lower accuracy. For tactile and EEG tasks, aligning with text produces the best results. Our speculation is that in the case of tactile data, the aligned images depict close-up views of objects, differing from those used in CLIP training. Consequently, the CLIP image encoder might not offer the optimal alignment space. As for EEG, due to the very limited scale of data, employing text-only alignment seems to be the most effective approach.

Anchor data ▼	MN40	SUN-D	ESC	TAG-M	IN-EEG
I	52.1	29.9	63.8	29.9	26.3
T	48.3	47.6	59.4	71.9	39.0
I+T	65.4	50.9	71.2	63.6	35.9

Table 19. **Align to different anchor data during training.** For different modalities, we show the classification results or zero-shot classification results when aligned to Image(I), Text(T) or Image and Text (I+T) during training.

B.8.2 Different Ratio of Training Data

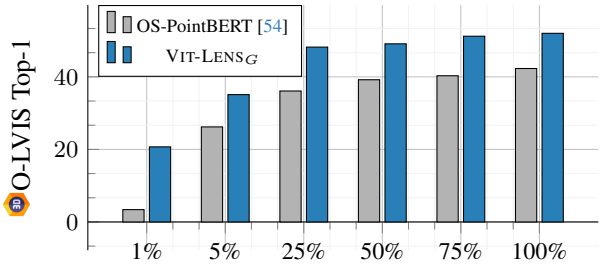


Figure 10. **Using different ratios of training data** in OpenShape Triplets to train for 3D point cloud. Zero-shot prediction on O-LVIS is reported for OS-PointBERT and VIT-LENS$_G$.

Our investigation delves into the influence of training data by performing ablation studies using different ratios of OpenShape Triplets [54] for training a 3D point cloud encoder. Specifically, we compare the performance of VIT-LENS$_G$

against OpenShape PointBERT in zero-shot classification on O-LVIS. The results, presented in Fig. 10, demonstrate that VIT-LENS$_G$ consistently outperforms OpenShape PointBERT across all ratios. Remarkably, in scenarios with limited training data (e.g., 1% training data), VIT-LENS showcases a significant performance advantage over PointBERT. This suggests the data-efficient nature of VIT-LENS, attributed to the rich knowledge encapsulated within pretrained-ViT.

B.8.3 Additional Architectural Ablations

This section focuses on additional ablation experiments centered around architectural designs. We focus on 3D tasks in this section. By default, our pretraining phase utilizes ULIP-ShapeNet Triplets [92], followed by evaluation on the ModelNet40 [90] benchmark on zero-shot classification.

Comparison with PointBERT. We conduct experiments to compare VIT-LENS with PointBERT [97], a transformer based architecture for 3D point cloud understanding. This comparison involves aligning to the feature space of different CLIP variants and employing distinct pretraining datasets (ULIP-ShapeNet, ULIP2-Objaverse and OpenShape Triplets). As is shown in Tab. 20, VIT-LENS outperforms PointBERT over all combinations of pretraining datasets and CLIP model for alignment. This substantiates the efficacy of harnessing a pretrained ViT to advance 3D shape understanding.

PT.D::CLIP Model ▼	PointBERT	VIT-LENS
▶:: OpenAI-B16	60.2	61.7
▶:: OpenCLIP-B16	62.6	65.4
▶:: OpenAI-L14	61.2	63.3
▶:: OpenCLIP-L14	65.4	70.6
▶:: OpenAI-B16	70.6	73.4
▶:: OpenCLIP-B16	71.7	74.8
▶:: OpenAI-L14	74.1	76.1
▶:: OpenCLIP-L14	77.8	80.6
▶:: OpenCLIP-bigG14	84.4	87.4

Table 20. **Comparisons with PointBERT.** We use different pretraining datasets(▶:ULIP-ShapeNet, ▶:ULIP-2 Objaverse, ▶:OpenShape Triplets) and different CLIP models as the foundation model for alignment. We report Top-1 accuracy on MN40.

Configuration of Iter-CA-Attn type Lens in VIT-LENS. We delve into the impact of different design choices of the Iter-CA-Attn type employed in VIT-LENS for 3D encoder. Our study encompasses the ablation of number of basis blocks (depth), as well as the exploration of parameter sharing beyond the second basis block (included), following [41]. The results outlined in Tab. 21 indicate that, beyond a certain threshold, notably four in our setting, increasing the number of basis blocks does not yield improvements in performance. Moreover, parameter sharing among blocks demonstrates its capability to reduce parameters while achieving comparable performance. This emphasizes the efficacy and efficiency of the Iter-CA-Attn Lens architecture within VIT-LENS for establishing connections between the 3D input and a pretrained-ViT.

Depth	Share Weights	#T.param	Acc@1
2	-	34.1M	64.8
4	✗	67.5M	64.2
4	✓	34.1M	65.4
6	✗	100.8M	65.1
6	✓	34.1M	64.0
8	✗	134.2M	64.0
8	✓	34.1M	64.3

Table 21. **Configuration of Iter-CA-Attn Lens on depth and parameters sharing.** We show the number of trainable parameters and report the zero-shot Top-1 accuracy on MN40. The default setting is marked with color.

Other hyper-parameters in VIT-LENS. We vary the number of latents used in the Lens of VIT-LENS$_B$. Note that the number of latents equals to the sequence length of the pretrained-ViT input. As delineated in Tab. 22, employing a larger number of latents, such as 384 and 512, shows slightly improved performance while concurrently increasing computational complexity measured in GFlops. This observation underscores the inherent capability of the CA-Iter-Attn type Lens to extract information from inputs of variable sizes and seamlessly connect them to the pretrained-ViT, mitigating computational complexity. Additionally, we investigate whether the inclusion of the pretrained-ViT position embedding influences model performance. Specifically, we interpolate the original position embedding while varying the number of latents. The results presented in Tab. 22 suggest that omitting the pretrained position embedding does not notably degrade performance. This suggests that the Lens is able to implicitly assimilate position information.

#latents	ViT.pos	Flops	Acc@1
128	✗	54.0G	65.1
128	✓	54.0G	65.2
196	✗	75.4G	65.1
196	✓	75.4G	65.4
256	✗	94.6G	65.5
256	✓	94.6G	65.5
384	✗	136.4G	66.2
384	✓	136.4G	66.3
512	✗	179.5G	66.3
512	✓	179.5G	67.4

Table 22. **Configuration of #latents and ViT position embedding.** We vary the number of latent queries and switching the use of the original pretrained-ViT position embeddings. The results showcase the corresponding GFlops to indicate computational complexity, along with reporting the Top-1 zero-shot accuracy on MN40. We show the default setting marked with color for clarity.

PointEmbed → Lens. To validate the efficacy of the pretrained-ViT, we investigate the performance of the "PointEmbed → Lens" paradigm. In this setup, the mean pooling feature of the CA-Iter-Attn Lens aligns directly with the CLIP feature space. We conduct experiments with various hyper-parameter configurations, and the comprehensive outcomes are presented in Tab. 23. Specifically, the configuration featuring a "depth of 6, with no parameter sharing" possesses a total parameter count comparable to the default setting of VIT-LENS (approximately 119M parameters). Despite having less trainable parameters, VIT-LENS outperforms this variant of "PointEmbed → Lens" by a significant margin. Besides, VIT-LENS also outperforms the rest variants. This observation underscores the importance of harnessing the capabilities of the pretrained-ViT.

Depth	#latents	Share Weights	#T.param	Flops	Acc@1
2	196	-	34.1M	27.4G	62.2
4	196	✗	67.5M	40.5G	62.4
8	196	✗	134.6M	66.7G	62.7
6	196	✗	101.2M	53.6G	61.9
6	196	✓	34.1M	53.6G	62.3
6	256	✗	101.3M	65.6G	63.5
6	256	✓	34.2M	65.6G	62.7
6	512	✗	101.5M	116.6G	62.5
6	512	✓	34.4M	116.6G	62.3
Default setting of VIT-LENS$_B$					
4	196	✓	34.1M	75.4G	65.4

Table 23. **Configurations for PointEmbed → Lens.** We vary the depth of Lens and alter sharing weights in Lens. We report the corresponding trainable parameters and zero-shot Top-1 accuracy on MN40. We show the default setting marked with color at the bottom for clarity.

PointEmbed → pretrained-ViT. We also delve into the paradigm of "PointEmbed → pretrained-ViT". As detailed in Tab. 24, training only the PointEmbed yields a zero-shot accuracy of 50%, significantly lower than that achieved by VIT-LENS due to the restricted number of trainable parameters. Subsequently, enabling the training of transformer blocks results in an improved zero-shot performance. However, this specialized training approach tailored specifically for enhancing 3D understanding might limit the adaptability of the resulting ViT to other modalities, potentially impacting the overall generalization ability of the ViT. In contrast, VIT-LENS achieves commendable performance while largely preserving the core parameters of the pretrained-ViT. This strategy effectively harnesses the extensive knowledge embedded within the pretrained-ViT across diverse modalities, with only a marginal increase in new parameters, showcasing its robustness and adaptability.

Unlocked Components in ViT	#T.param	Flops	Acc@1
None	7.3K	111.4G	50.0
`[CLS]`	7.3K	111.4G	53.6
`[CLS], Proj`	1.1M	111.4G	60.8
`[CLS], Proj, Block.1, Block.2`	15.3M	111.4G	64.8
`[CLS], Proj, Block.11, Block.12`	15.3M	111.4G	64.2
`[CLS], Proj, Block.1 - Block.4`	29.5M	111.4G	65.4
`[CLS], Proj, Block.9 - Block.12`	29.5M	111.4G	64.7
`[CLS], Proj, Block.1 - Block.6`	43.7M	111.4G	66.4
`[CLS], Proj, Block.7 - Block.12`	43.7M	111.4G	65.6
All	86.6M	111.4G	67.7
Default setting of VIT-LENS$_B$			
None(tune PointEmb, Lens)	34.1M	75.4G	65.4

Table 24. **Configurations for PointEmbed→pretrained-ViT.** We vary the sub-modules of pretrained-ViT unlocked during training. We report the corresponding trainable parameters, GFlops and zero-shot Top-1 accuracy on MN40. We show the default setting marked with color at the bottom for clarity.

C. Further Discussion

Beyond using pretrained-ViT. The core of VIT-LENS in advancing representations across diverse modalities relies on leveraging the profound knowledge embedded within the pretrained-ViT. Given the significant enhancements facilitated by the pretrained-ViT, an initial exploration involves employing the powerful Large Language Model (LLM) to encode inputs across various modalities. In this endeavor, we replace the pretrained-ViT with Flan-T5 XL [13] within the VIT-LENS architecture. To facilitate alignment, we introduce an additional trainable token. Training the model on ULIP-ShapeNet and ULIP2-Objaverse under various experimental configurations, we report the zero-shot classification performance on MN40. Results are show in Tab. 25. Notably, when trained on ULIP-ShapeNet, the model exhibits proficient alignment with CLIP (I+T), achieving a notable top-1 zero-shot accuracy of 62.6% on MN40. Moreover, upon scaling the model to the ULIP2-Objaverse dataset enriched with textual captions, a remarkable improvement is observed. Specifically, it achieves an outstanding top-1 accuracy of 79%, surpassing the performance obtained by training PointBERT from scratch with the same CLIP model for alignment. This outcome underscores the potential of this approach for omni-modal learning. We leave further exploration of this promising avenue to future work.

Pretrained Data	Align to	Acc@1
ULIP-ShapeNet	OpenCLIP-L14 (T)	48.7
ULIP-ShapeNet	Flan-T5 (T)	52.5
ULIP-ShapeNet	OpenCLIP-L14 (I+T)	62.6
ULIP2-Objaverse	OpenCLIP-L14 (T)	68.2
ULIP2-Objaverse	Flan-T5 (T)	72.2
ULIP2-Objaverse	OpenCLIP-L14 (I+T)	79.0

Table 25. **Train 3D encoder with pretrained Flan-T5 XL.** We use different pretrained data and foundation modelS for alignment. We report zero-shot Top-1 accuracy on MN40.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv*, 2022. 2, 3, 4

[2] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017. 2

[3] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. Multimae: Multi-modal multi-task masked autoencoders. In *ECCV*, 2022. 6

[4] Yunpeng Bai, Xintao Wang, Yanpei Cao, Yixiao Ge, Chun Yuan, and Ying Shan. Dreamdiffusion: Generating high-quality images from brain eeg signals. *arXiv*, 2023. 1, 2, 5, 6, 11

[5] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv*, 2021. 2

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020. 2

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2, 3, 8

[8] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv*, 2015. 5, 9, 10

[9] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP*, 2020. 4, 11, 12

[10] Sihan Chen, Handong Li, Qunbo Wang, Zijia Zhao, Mingzhen Sun, Xinxin Zhu, and Jing Liu. Vast: A vision-audio-subtitle-text omni-modality foundation model and dataset. *arXiv*, 2023. 6

[11] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme, Andreas Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. Pali: A jointly-scaled multilingual language-image model. *arXiv*, 2022. 2

[12] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. *arXiv*, 2022. 2, 3, 4, 5, 6

[13] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv*, 2022. 21

[14] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *CVPR*, 2022. 5, 10

[15] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR Workshop*, 2020. 11

[16] 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. *arXiv*, 2023. 2, 4, 7, 13

[17] 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, 10

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*. 11

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv*, 2018. 2, 3

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv*, 2020. 2

[21] 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. *arXiv*, 2023. 2

[22] Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. Clotho: An audio captioning dataset. In *ICASSP*, 2020. 4, 10

[23] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *arXiv*, 2023. 2

[24] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *CVPR*, 2023. 2, 3, 7, 13

[25] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. In *IJCV*, 2021. 5, 10

[26] Difei Gao, Ke Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. Multi-modal graph neural network for joint reasoning on vision and scene text. In *CVPR*, 2020. 2

[27] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv*, 2023. 2, 3, 4, 7, 13

[28] Yuying Ge, Sijie Zhao, Ziyun Zeng, Yixiao Ge, Chen Li, Xintao Wang, and Ying Shan. Making llama see and draw with seed tokenizer. *arXiv*, 2023. 2, 3, 4, 7, 13

[29] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017. 4, 5, 10, 12

[30] Amin Ghiasi, Hamid Kazemi, Eitan Borgnia, Steven Reich, Manli Shu, Micah Goldblum, Andrew Gordon Wilson, and Tom Goldstein. What do vision transformers learn? a visual exploration. *arXiv*, 2022. 9

[31] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *CVPR*, 2022. 2, 6, 9

[32] 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. 1, 2, 3, 5, 6, 8, 9, 10, 11

[33] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Omnimae: Single model masked pretraining on images and videos. In *CVPR*, 2023. 2

[34] Yuan Gong, Yu-An Chung, and James Glass. AST: Audio Spectrogram Transformer. In *Interspeech*, 2021. 5, 9

[35] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv*, 2021. 2

[36] 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*, 2023. 3

[37] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 10

[38] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. Audioclip: Extending clip to image, text and audio. In *ICASSP*, 2022. 2, 3, 4, 6

[39] Jiaming Han, Renrui Zhang, Wenqi Shao, Peng Gao, Peng Xu, Han Xiao, Kaipeng Zhang, Chris Liu, Song Wen, Ziyu Guo, et al. Imagebind-llm: Multi-modality instruction tuning. *arXiv*, 2023. 3

[40] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 2, 3, 6

[41] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *ICML*, 2021. 4, 12, 20

[42] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 2

[43] Micah K. Johnson and Edward H. Adelson. Retrographic sensing for the measurement of surface texture and shape. In *CVPR*, 2009. 5, 9

[44] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *NAACL-HLT*, 2019. 4, 11

[45] Weixian Lei, Yixiao Ge, Jianfeng Zhang, Dylan Sun, Kun Yi, Ying Shan, and Mike Zheng Shou. Vit-lens: Towards omni-modal representations. *arXiv*, 2023. 1

[46] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 2

[47] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 3, 10

[48] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv*, 2023. 3, 4, 10

[49] 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*, 2023. 3

[50] Valerii Likhosherstov, Anurag Arnab, Krzysztof Choromanski, Mario Lucic, Yi Tay, Adrian Weller, and Mostafa Dehghani. Polyvit: Co-training vision transformers on images, videos and audio. *arXiv*, 2021. 2

[51] Kevin Qinghong Lin, Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Z XU, Difei Gao, Rong-Cheng Tu, Wenzhe Zhao, Weijie Kong, et al. Egocentric video-language pretraining. *NeurIPS*, 2022. 2

[52] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. 2023. 3, 4

[53] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv*, 2023. 2, 3, 4

[54] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. *arXiv*, 2023. 1, 2, 4, 5, 6, 7, 8, 10, 11, 13, 14, 15, 19

[55] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv*, 2019. 2, 3

[56] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv*, 2022. 5

[57] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 6

[58] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv*, 2021. 13, 14, 15

[59] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In *CVPR*, 2021. 2

[60] Arsha Nagrani, Paul Hongsuck Seo, Bryan Seybold, Anja Hauth, Santiago Manen, Chen Sun, and Cordelia Schmid. Learning audio-video modalities from image captions. In *ECCV*, 2022. 6

[61] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 4, 10

[62] Andreea-Maria Oncescu, A Koepke, Joao F Henriques, Zeynep Akata, and Samuel Albanie. Audio retrieval with natural language queries. *arXiv*, 2021. 11

[63] OpenAI. Introducing chatgpt. OpenAI Blog, 2021. 2, 3

[64] OpenAI. Gpt-4 technical report, 2023. 10

[65] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv*, 2023. 2, 3

[66] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv*, 2019. 11

[67] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F. Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *ICLR*, 2021. 6

[68] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023. 2, 19

[69] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv*, 2022. 2

[70] Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *ACM MM*, 2015. 4

[71] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 9

[72] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 2017. 5

[73] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *NeurIPS*, 2022. 5

[74] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI blog*, 2018. 2, 3

[75] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019. 2, 3

[76] 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. 2, 3, 4

[77] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *NeurIPS*, 2021. 9

[78] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, 2015. 4, 5, 10

[79] Concetto Spampinato, Simone Palazzo, Isaak Kavasidis, Daniela Giordano, Nasim Souly, and Mubarak Shah. Deep learning human mind for automated visual classification. In *CVPR*, 2017. 4, 5, 11

[80] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv*, 2023. 3

[81] 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. *arXiv*, 2023. 3

[82] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020. 6

[83] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020. 2

[84] 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*, 2023. 2, 3

[85] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *ICCV*, 2019. 4, 10

[86] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 3

[87] Peng Wang, Shijie Wang, Junyang Lin, Shuai Bai, Xiaohuan Zhou, Jingren Zhou, Xinggang Wang, and Chang Zhou. One-peace: Exploring one general representation model toward unlimited modalities. *arXiv*, 2023. 2, 3, 6

[88] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. *arXiv*, 2023. 3

[89] Yusong Wu*, Ke Chen*, Tianyu Zhang*, Yuchen Hui*, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, 2023. 6

[90] 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. 4, 10, 20

[91] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. *CVPR*, 2016. 6

[92] 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. 2, 4, 5, 9, 11, 20

[93] 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. *arXiv*, 2023. 2, 4, 5, 10, 11

[94] Fengyu Yang, Chenyang Ma, Jiacheng Zhang, Jing Zhu, Wenzhen Yuan, and Andrew Owens. Touch and go: Learning from human-collected vision and touch. In *NeurIPS*, 2022. 1, 2, 4, 5, 6, 11

[95] Kim Youwang, Kim Ji-Yeon, and Tae-Hyun Oh. Clip-actor: Text-driven recommendation and stylization for animating human meshes. In *ECCV*, 2022. 2

[96] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv*, 2022. 2

[97] 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. 5, 9, 10, 13, 20

[98] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *CVPR*, 2022. 2

[99] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv*, 2017. 11

[100] Jianfeng Zhang, Hanshu Yan, Zhongcong Xu, Jiashi Feng, and Jun Hao Liew. Magicavatar: Multi-modal avatar generation and animation. In *arXiv*, 2023. 2

[101] 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. 2, 5

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

[103] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020. 11

[104] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 2

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

[106] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyao Zeng, Shanghang Zhang, and Peng Gao. Pointclip v2: Adapting clip for powerful 3d open-world learning. *arXiv*, 2022. 2, 5