GiT: Towards Generalist Vision Transformer through Universal Language Interface

Haiyang Wang[1,2*] Hao Tang[1*] Li Jiang[2✉] Shaoshuai Shi[2]
Muhammad Ferjad Naeem[3] Hongsheng Li[4] Bernt Schiele[2] Liwei Wang[1,5✉]

[1]Center for Machine Learning Research, Peking University
[2]Max Planck Institute for Informatics, Saarland Informatics Campus
[3]ETH Zurich [4]The Chinese University of Hong Kong
[5] State Key Laboratory of General Artificial Intelligence, SIST, Peking University

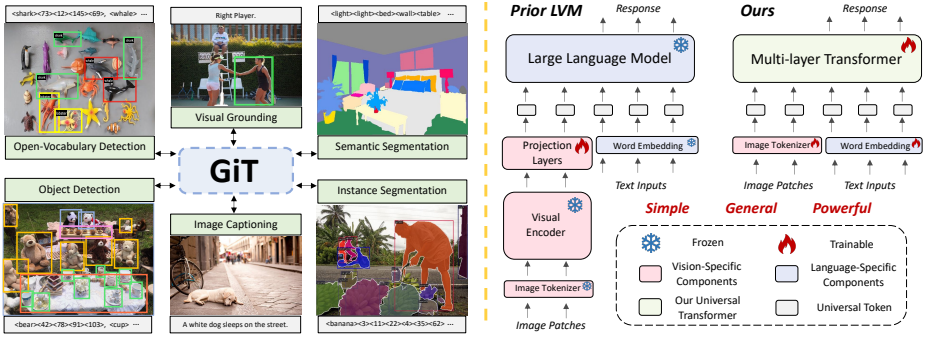

Fig. 1: ***G****eneralist* *Vi****s****ion* ***T****ransformer*. a) Examples of tasks supported by GiT. b) Architectural comparison between previous LVMs (*e.g.*, LLaVA [48]), and ours. GiT seamlessly handles various vision-centric tasks, particularly fine-grained visual perception, via a universal language interface using a plain transformer (*e.g.*, ViT and GPT).

Abstract. This paper proposes a simple, yet effective framework, called GiT, simultaneously applicable for various vision tasks only with a vanilla ViT. Motivated by the universality of the Multi-layer Transformer architecture (*e.g.*, GPT) widely used in large language models (LLMs), we seek to broaden its scope to serve as a powerful vision foundation model (VFM). However, unlike language modeling, visual tasks typically require specific modules, such as bounding box heads for detection and pixel decoders for segmentation, greatly hindering the application of powerful multi-layer transformers in the vision domain. To solve this, we design a universal language interface that empowers the successful auto-regressive decoding to adeptly unify various visual tasks, from image-level understanding (*e.g.* captioning), over sparse perception (*e.g.* detection),

* Equal contribution. ✉ Corresponding author.

to dense prediction (*e.g.* segmentation). Based on the above designs, the entire model is composed solely of a ViT, without any specific additions, offering a remarkable architectural simplification. GiT is a multi-task visual model, jointly trained across five representative benchmarks without task-specific fine-tuning. Interestingly, our GiT builds a new benchmark in generalist performance, and fosters mutual enhancement across tasks, leading to significant improvements compared to isolated training. This reflects a similar impact observed in LLMs. Further enriching training with 27 datasets, GiT achieves strong zero-shot results over various tasks. Due to its simple design, this paradigm holds promise for narrowing the architectural gap between vision and language. Code and models are available at https://github.com/Haiyang-W/GiT.

Keywords: Unified Visual Modeling · Multi-Task Learning

1 Introduction

Developing a universal model capable of completing various tasks aligned with human intention is a long standing goal in Machine Learning. In language processing, the emergence of LLMs [1, 60, 71, 89] opens up a promising route, which only employs several stacked transformer layers for adaptable task management with minimal prompts. In this paper, we explore this simple multi-layer transformer [73] architecture in visual modeling, seamlessly integrating numerous vision tasks with a universal language interface, aiming to close the architecture gap between vision and language.

The Machine Learning community is undergoing a paradigm shift with the rise of foundation models (*e.g.*, GPT [9], BERT [37], DALL-E [62]) trained on massive data, enabling the sharing of conceptual knowledge, and offering seamless adaptability to diverse downstream tasks. Language models [9, 37, 71] have greatly benefited from this recently, thanks to a homogenized representation (*i.e.*, input and output are uniformly represented as token sequence). State-of-the-art models like GPT4 [56], LLaMA [71], PaLM2 [1] and Gemini [70] have shown an unprecedented ability to follow human instructions and solve open-ended tasks. Thanks to their success, this architecture is potentially viewed [8,63] as a general framework for other machine learning tasks beyond NLP.

Motivated by this opportunity, the community has developed several large vision models, such as LLaVA [48], Unified-IO [52] and OFA [77], by leveraging vision features [28, 34] as foreign language of open-source LLMs [61, 69, 71]. However, this progress still retained task-specific designs, including diverse visual encoders [77,91], perception heads [42], RPN [42], and specific target representations [52]. Task-specific modules require intricate designs for each task a model needs to solve, potentially hindering progress towards a general vision model. Moreover, these task-specific designs typically involve numerous separate training stages [78], complicating model scaling across different tasks. We argue that an alternative general-purpose framework could employ lightweight components through a more universal input-output interface, and allocate most of the model resources to learning a general model across these tasks.

Table 1: Columns from left to right display task source examples, dataset counts, total samples, percentages, and multi-task sampling rates, then task modalities. Highlighted rows summarize statistics for similar task groups. See appendix for the complete list.

	Example Sources	Size				Input Modalities		Output Modalities		
		Dataset	Size	Percent	Weight	Text	Image	Text	Sparse	Dense
Image-Level		**10**	**11.4m**	**67.1**	**40**	✓	✓	✓	✓	-
Image Captioning	*CC12M [13], VG [40], SBU [57]*	5	11.3m	66.6	30	-	✓	✓	-	-
Visual Grounding	*RefCOCO [87], Flickr30k [59]*	5	115k	0.7	10	✓	✓	-	✓	-
Object-Level		**11**	**5.2m**	**30.9**	**40**	-	✓	-	✓	✓
Object Detection	*Objects365 [66], COCO [47]*	8	3.8m	22.6	20	-	✓	-	✓	-
Instance Segmentation	*OpenImages [41], LVIS [31]*	4	1.4m	7.9	20	-	✓	-	✓	✓
Pixel-Level		**6**	**322k**	**2.0**	**20**	-	✓	-	-	✓
Semantic Segmentation	*COCOStuff [11], ADE20K [90]*	6	322k	2.0	20	-	✓	-	-	✓
All Tasks		**27**	**17m**	**100**	**100**	✓	✓	✓	✓	✓

Previous attempts [3, 7, 26, 44, 48, 79, 91] on large visual modeling predominantly focused on the image-level vision-language domain, mainly due to its straightforward integration into LLMs by viewing the image as a foreign language. This approach often overlooks the incorporation of classical perception tasks, such as detection and segmentation. Developing a unified framework for fundamental visual perception has proven to be quite challenging since it requires the model to predict multiple outputs with different formats in parallel, with annotations varying widely in representations, ranging from coarse-grained image level to fine-grained pixel level. For example, detection yields variable numbers of bounding boxes, segmentation produces binary masks, and image captioning generates textual answers. These drawbacks make it difficult to design a single model simultaneously applicable across all visual tasks.

Recent developments in LLMs [4,9,55,56] have shown the potential of Transformer [73] being a universal computation architecture. Inspired by this, we introduce GiT, a vision foundation model that can handle diverse vision-centric tasks. As shown in Figure 1, compared to previous unified models [52, 77, 78], our method features a minimalist design, comprising just several Transformer layers without any vision-specific additions other than the patch projection layers, closely aligning with LLM architectures. Similar to language modeling, all visual tasks are structured into an auto-regressive framework through a universal language interface. Specifically, our targets are expressed as token sequences using a unified representation, relying solely on a standard vocabulary without involving extra tokens [63, 78]. To be compatible with various visual tasks across different perceptual scales, we introduce a flexible multi-task template for task unification. It partitions the whole image into N subregions by grid sampling and concurrently processes each subregion with efficient parallel decoding.

The above designs facilitate multi-task training of our model across five representative benchmarks without task-specific fine-tuning. As shown in Table 3 and 4, leveraging shared parameters and representation, our model achieves strong generalist results and mirrors the multi-task capabilities of LLMs [4]. Tasks with overlapping abilities can mutually enhance each other, leading to significant gains over separate training (see §5.2 for more analysis). To further enhance general-

izability, we incorporate 27 standard visual datasets into training (see Table 1), resulting in strong zero- and few-shot performances on unseen data.

In particular, our work makes the following contributions:

- *Foundational framework for unified visual modeling.* We introduce a simple visual modeling paradigm with a straightforward multi-layer transformer, greatly simplifying the model design. Our model integrates various vision-centric tasks, especially the often-neglected object- and pixel-level tasks, via an efficient universal language interface.
- *Multi-task ability like LLMs.* Weight-sharing and unified learning objectives enable us to obtain the multi-task capability as observed in LLMs, achieving the best and mutually enhanced generalist performance over five benchmarks.
- *Strong generalizability.* Fully embracing the one-stage joint training strategy as used in LLMs, our model is trained on 27 publicly available datasets, achieving strong zero- and few-shot performance across various tasks.

2 Related Work

Multi-layer Transformer [73] has emerged as a universal learning architecture, becoming a cornerstone in most LLM frameworks. Notable LLMs like GPT series [4, 9, 55, 56, 58, 60], as well as LLaMA [71], PaLM [1], and OPT [89] have made significant advances in this domain. Beyond language, plain transformer also has proven effective in 2D vision with ViT [28], 3D vision via DSVT [74], multimodal imaging in UniTR [75]. Despite their success, these straightforward transformers are often limited to feature encoding and require task-specific modules, greatly hindering the progress toward a general learner. To solve this, we aim to broaden the scope of multi-layer transformer, moving beyond their conventional encoder-only function to an LLM-like visual modeling, narrowing the architectural gap between the vision and language.

Vision Foundation Model excels in handling diverse visual tasks within a unified architectural framework. Motivated by the success of seq2seq models in NLP, innovations like Flamingo [3], LLaVA [48] and Gato [63] have reframed vision-language tasks as sequence generation problems, which is further developed by Pix2Seq v2 [20], and VisionLLM [78] to process spatial information across more tasks. However, these methods face challenges such as inefficient inference from non-parallel decoding [20], negative transfer [92] or complex vision-specific additions [42, 52, 78], slowing progress towards a universal vision model.

3 Universal Language Interface

In this section, we propose a simple universal language interface that integrates five fundamental visual tasks, ranging from image, over object to the pixel level, into the successful auto-regressive framework. All our targets are expressed as token sequences via a unified representation (§3.1), and then organized by a general

Table 2: Summary of architecture configuration. Shared parameters account for over 98% of the whole model. The parameter of text embedding is excluded because it operates in a zero-computation index manner.

Model	Multi-Modal Tokenizers			Multi-layer	Layer	Total
	Text	Image	Out-of-vocab	Transformer	Number	Parameter
GiT $_{\text{Base}}$	0	0.4%	1.8%	97.8%	18 (12+6)	131M
GiT $_{\text{Large}}$	0	0.2%	1.1%	98.7%	30 (24+6)	387M
GiT $_{\text{Huge}}$	0	< 0.1%	0.8%	99.1%	38 (32+6)	756M

Table 3: Abilities required for each task and the performance improvements after multi-task training. † means polygon-based segment [82, 83], different from the popular mask-based methods [32].

Task	Image	Language	Segment	Localization	*Improve* (single→multi)
Detection	✓	-	-	✓	+***1.6***@AP
InsSeg	✓	-	✓†	✓	+***1.6***@AP$_{50}$, +***0.2***@AP$_{75}$
Grounding	✓	✓	-	✓	+***2.5***@Acc
Caption	✓	✓	-	-	+***4.7***@CIDEr
SemSeg	✓	-	✓	-	+***0.1***@mIoU

multi-task template (§3.2), which partitions the fine-grained visual perception into a series of parallel-decoded subproblems. Figure 2 illustrates the multi-task input templates for three tasks, namely image captioning (image-level task, left), object detection (object-level task, middle) and semantic segmentation (pixel-level task, right). Further technical details are provided below.

3.1 Unified Input and Output Representation

To support various modalities such as images, language, bounding boxes, masks, *etc*, it's essential to represent them in a unified space. To achieve this, we straightforwardly project the input image and text into patch and language token embeddings. Following this, all targets are represented via a universal language interface and tokenized entirely based on a standard vocabulary [81].

Text representation. Vision-language tasks often require text processing, like image captioning, where a natural language description is generated based on the given image. To handle it, we follow the practice of BERT [37], texts are transformed into WordPiece [81] subwords, with a ∼30,000 token vocabulary, and then embedded via a lookup table into a learnable embedding space. Position encodings are added to indicate local positions within time steps.

Out-of-vocabulary representation. Visual perception typically relies on complex textual concepts comprised of multiple pieces, such as "traffic light" and "20 47", the category name and numerical value used in object detection. As discussed in [45, 78], using multiple tokens to represent them is inefficient. 1) Adding separators like </c>"traffic light"</c> to identify categories will extend sequence length, particularly impractical for dense prediction tasks. 2) Varying token length for multi-piece words leads to inconsistent decoding steps, necessitating complex and rule-based post-processing to achieve reliable outcomes. To tackle this problem, some solutions [52, 63, 78] introduce new tokens of category and number terms while facing challenges when considering token capacity constraints. Instead of expanding the vocabulary, we treat multi-piece concepts as continuous text and compress them into a single token as follows,

$$\begin{aligned} &\mathcal{I}_0, \mathcal{I}_1 = \text{Tokenizer}(\text{“traffic light”}), && \mathcal{I} \text{ is the token index,} \\ &\mathcal{F}_0, \mathcal{F}_1 = \text{Attention}(\text{TE}(\mathcal{I}_0) + \text{PE}(0), \text{TE}(\mathcal{I}_1) + \text{PE}(1)), && \mathcal{F}_{\text{traffic light}} = \mathcal{F}_0, \end{aligned} \tag{1}$$

where Attention$(\cdot)$ is a single-layer attention, TE$(\cdot)$ and PE$(\cdot)$ are text and position embedding functions. Our approach offers an alternative solution for

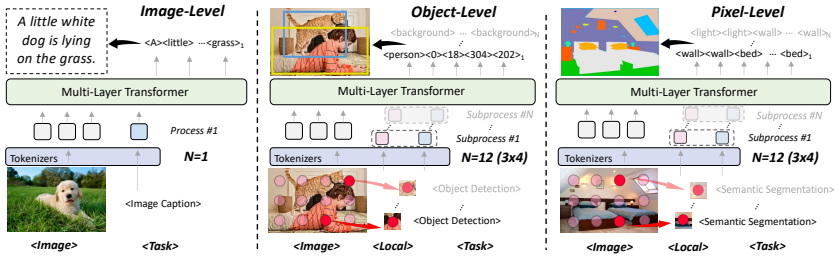

Fig. 2: Task-level customization spans from image- to object- and pixel-level, setting N to 1, 625 (25×25), and 1764 (42×42), in real implementation. Red point means localized visual token, generated by image bilinear interpolation at its grid point. Task prompt is text, converted into a token via text and out-of-vocabulary representation.

handling any out-of-vocabulary terms without expanding the basic vocabulary, which greatly simplifies the post-processing for achieving effective perception.

Sparse representation. In the context of sparse object-level perceptions such as object detection [12, 30] and instance segmentation [33], which generate various category and location representations (for example, bounding boxes and instance masks), we propose a standardized output format. This format is defined as a tuple (C, P), where C represents the category label, and $P{=}\{x_i, y_i\}_{i=1}^{N}$ denotes a set of N points that identify the object's location. To align with the format of linguistic tokens, both the class and location targets are tokenized by the prior text and out-of-vocabulary representation. Following VisionLLM [78], continuous coordinates are uniformly discretized into integers within `[-range, range]`. A bounding box is formulated with four points as $\{x_1, y_1, x_2, y_2\}$, representing its top-left and bottom-right coordinates, while instance mask defines its fine-grained region via multiple points along the boundary [82, 83].

Dense representation. Various perceptual tasks, such as semantic segmentation [51, 65], require models to generate dense outputs, often involving per-pixel predictions. To handle these tasks, we start by converting per-pixel labels into unified tokens. For example, semantic classes [47] are firstly tokenized by text and out-of-vocabulary representation. Then, these dense labelings are flattened into a 1D sequence in raster order, represented autoregressively, similar to iGPT [18].

Image representation. Images are converted into a non-overlapping 16 × 16 patch sequence in raster order and then embedded to tokens with a trainable linear projection and a learnable positional embedding, as done in ViT [28].

3.2 Multi-Task Template with Parallel Decoding

Prior to constructing the templates, we first divide 2D visual understanding into three distinct categories, each defined by their perceptual granularity and output representation. Our focus encompasses five core tasks for training and analysis: 1) *Image-level* tasks, exemplified by Image Captioning and Visual Grounding, 2) *Object-Level* tasks like Object Detection and Instance Segmentation, and 3)

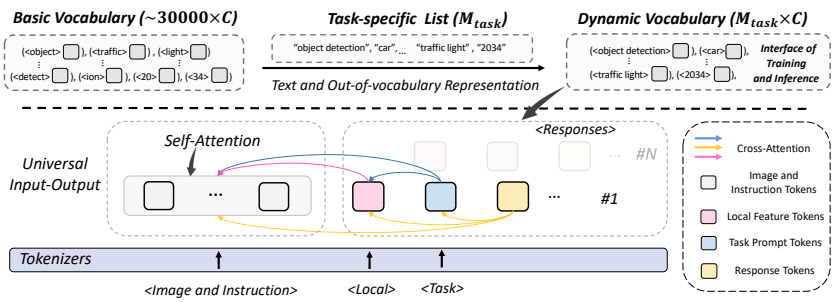

Fig. 3: Our multi-task formulation is broadly illustrated as processing four types of user inputs: image patches, instructive language tokens, and N parallel point-based subprocesses, each with its interpolated local image feature and task identifier for efficient parallel visual prediction. As for the language interface, we use a basic vocabulary, a specific vocabulary list required by the current task, and the task-agnostic out-of-vocabulary module (§3.1) to dynamically create vocabulary sets for each task.

Pixel-Level tasks such as Semantic Segmentation. Then, we introduce a unified seq2seq framework that seamlessly integrates various task formulations, from purely visual to those involving language, enabling flexible task customization.

General Formulation. Inspired by well-established language models, we adapt the widely accepted instruction template of LLMs to the vision community (*e.g.*, vision-language and spatial-aware visual perception). As shown in Figure 2 and 3, the instructional template is defined as follows,

$$\underbrace{\text{<Image><Instruction>}}_{\text{shared global observation}} \underbrace{\begin{cases} \text{<LocalFeature}_1\text{><Task}_1\text{>:<Response}_1\text{>} \\ \vdots \quad\quad \vdots \quad\quad \vdots \\ \text{<LocalFeature}_N\text{><Task}_N\text{>:<Response}_N\text{>.} \end{cases}}_{\text{multi-track local observations and responses}} \tag{2}$$

In our template, user input is structured into four segments. The first comprises image patches, as done in ViT. The second involves instruction inputs, like language expression used for visual grounding. For the third and fourth segments, targeting efficient object- and pixel-level visual perception like simultaneously predicting multiple bounding boxes as in traditional object detection, we partition the task into N parallel local subprocesses by grid sampling, as shown in Figure 2. Each subprocess works with a local image token, created by bilinearly interpolating image features based on its grid point position, and a pure text task identifier, converted into a single token via text and out-of-vocabulary representation. For Vision-Language tasks, we set N to 1, while for vision-centric tasks like detection and segmentation, N is adjustable to match the required prediction resolution. These designs allow our method to flexibly handle nearly all 2D vision tasks. Notably, some segments are optionally required by different tasks, *e.g.*, image captioning only requires image inputs and a task prompt.

In contrast to the traditional encoder and decoder setups, we employ var-

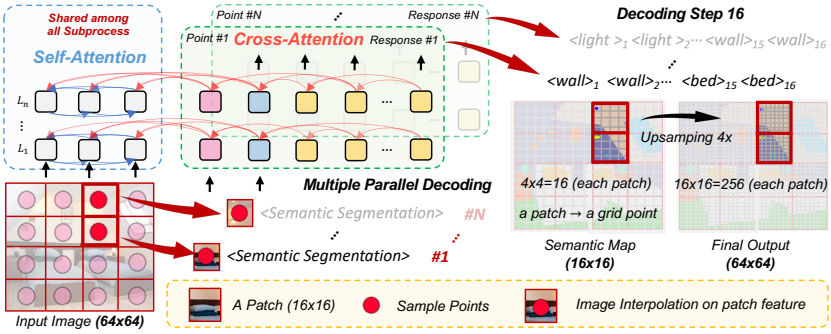

Fig. 4: An illustration of pixel-level multiple parallel decoding. Consider a 64×64 image divided into 16 patches, where each patch is 16×16. With N=16 and a decoding step of 16 per subprocess, each grid point covers one patch to predict a 4×4 semantic map, which is then upsampled 4× to the original size for the final result.

ious mask matrices to determine the token representation context. As shown in Figure 3, our method processes inputs (*i.e.*, image and instruction) by applying bidirectional self-attention, similar to a typical encoder. Importantly, we enable image-to-text attention to enhance its ability of text-conditioning image processing (see Table 7). As for computing local and task prompts, and target prediction of each subprocess, we use left-to-right unidirectional attention for modeling causal relations, in line with decoder-only autoregressive approach.

Image-Level. The definition for image-level tasks such as image captioning and visual grounding is straightforward, closely mirroring the NLP tasks. Following previous vision-language methods, we set N to 1 and structure the token sequence of image captioning as {<image> "*image captioning*": <text>}, and visual grounding as {<image> <instruction> "*visual grounding*": <bbox>}.

Object-Level. Developing a generative framework that adeptly manages classical object-level perception tasks, including object detection and instance segmentation, presents a significant challenge. It demands a model capable of concurrently generating all the bounding boxes and masks. To address this, as shown in Figure 2, we introduce a point-based parallel decoding framework designed for visual prompt perception. It starts by sampling a grid of N points across the image, where N is set to 625, corresponding to a 25 × 25 sampling resolution for 1120 × 1120 images. Following this, we conduct generative perception at each point using the format: {<image> <local feature> <task identifier>: <sparse response>}. <image> is the patch tokens shared by all grid subprocesses. <sparse response> indicates our chosen object-level sparse representation as detailed in §3.1. Notably, if the point is in the negative part, <background> token will be predicted.

An example of detection for a grid point: {<image><local feature> "*object detection*": <c><x_1><y_1><x_2><y_2>}, where <c> is the class label, and (<x_1><y_1><x_2><y_2>) indicate the box points' offsets from the grid points.

Pixel-Level. The auto-regressive decoding paradigm [9, 56, 60] struggles with high-dimensional outputs, particularly in cases like computing all pixel semantic categories in a single sequence, incurring considerable computational overhead. Earlier efforts [52, 54] attempted to alleviate this using compressed tokens via VQ-VAE [72]. However, this approach compromised the pure language interface and introduced intricate modules. To tackle this issue, as illustrated in Figure 4, we convert per-pixel labels into linguistic tokens and further divide the image into N uniform sub-regions, just like object-level tasks. Specifically, for segmentation tasks, we set N to 1764 to achieve a 42×42 perceptual resolution for images sized 672 × 672. Each subprocess independently conducts sequential pixel-level predictions in parallel, leading to enhanced efficiency.

An example of semantic segmentation for a single track with 16 decoding steps: {<image> <local feature> "*semantic segmentation*": <c_1> <c_2> ⋯ <c_{15}> <c_{16}>}, where <c_i > is the i-th class token of each sub-region.

4 Training

4.1 Architecture: Multi-layer Transformer

By employing the universal language interface, we formulate a diverse array of 2D vision tasks as sequences of discrete input and output tokens. This method has paved the way for extending the successful architectures (such as Multi-layer Transformers [9, 60, 73]) in Large Language Models, to unified visual modeling.

Building on the visual foundations, we leverage the structure of window-based ViT [28, 46], identical to the visual encoder used in SAM [39], for both linguistic sequences and high-resolution images. A few global attention blocks are evenly integrated into the model for feature propagation. Notably, within the window attention layer, each patch token only interacts with grid points located in the same window. Our approach can be built upon such a common structure (*i.e.*, ViT) without architectural changes, enhancing the framework's universality.

Benefiting from the above designs, our architecture can allocate the most of computational parameters (> 98%) to general inference, complemented by a few lightweight modules for diverse modality inputs, as shown in Table 2.

4.2 Multi-Task and Universal Training

GiT undergoes joint training across various tasks and datasets. Our goal is to assess the capability of a unified model to handle multiple tasks simultaneously. Thus, we refrain from task-specific fine-tuning, despite prior studies demonstrating its potential to enhance task performance.

Various Tasks and Datasets. To build a singular unified model for diverse perception and V&L tasks, we construct an analyzable multi-task benchmark comprising the most representative datasets across five fundamental tasks we previously identified, spanning from image- to pixel-level visual understanding. To enhance the model's adaptability, we augment the benchmark by integrating

27 datasets from 16 publicly accessible data sources, as listed in Table 1.

Joint Multi-Task Training. We jointly train GiT on the above multi-task benchmark by mixing samples from these datasets. As detailed in Table 1, to prevent overshadowing tasks with smaller data during joint training and avoid potential performance drops, we uniformly sample from all tasks (1/5), regardless of their data sizes. In universal settings where tasks span multiple domains, sampling inside each task is balanced across scenarios like daily life, indoor, and outdoor. Within these domains, datasets are sampled in proportion to their size.

Regarding the learning objective, different tasks require distinct vocabularies. For example, visual grounding uses numerical coordinates, whereas segmentation involves semantic concepts. To tackle this problem, as illustrated in Figure 3, we approach all tasks as the next token generation problem using standard CrossEntropy loss, while employing a task-specific vocabulary. This allows for dynamically controlling vocabulary sets, adapting to the unique requirements of each task during both training and inference phases.

Scaling Models. We adopt a variant of ViT [28] similar to SAM [39], augmented with six extra transformer layers and text embeddings used in BERT [37] to improve non-visual modality processing (refer to appendix). To study the dependence of performance on model scale, we introduce three different sizes of model built up on ViT-B, -L, and -H, with parameters ranging from 131M to 756M, detailed in Table 2. The initial layers inherit parameters pretrained by SAM, while the new layers start with random initialization.

5 Experiments

5.1 Experimental Settings

Multi-Task Datasets. To facilitate in-depth analysis and fair evaluation, we built an analyzable multi-task benchmark, choosing one of the most representative datasets for each task. To ensure consistency and enable comparison with VisionLLM [78], we retained the same datasets they used for the four vision-centric tasks: COCO2017 [47] for object detection and instance segmentation, COCO Caption [21] for image captioning, and the RefCOCO series [53, 87] for visual grounding. For the semantic segmentation not included in VisionLLM, we employed the widely used ADE20K dataset [90].

Extended Datasets. To showcase the universality of our unified framework, we enhanced our multi-task benchmark by integrating more standard and publicly available datasets from vision-language and visual perception (see §4.2).

Training and Evaluation Details. To illustrate the flexibility and efficacy of our model, we established three training paradigms: single-task, multi-task, and universal setting. In single-task training, the focus is on optimizing performance on individual benchmarks. Multi-task training, on the other hand, targets the development of a general learner across five selected datasets. Drawing from the insights in Uni-Perceiver v2 [42], we adopt an unmixed sampling strategy (*i.e.*, sampling one task per iteration) for faster and more stable training, However, our framework is also compatible with in-batch mixing strategies [52,94] as suggested

Table 4: Results on standard vision-centric benchmarks. "single-task" refers to models trained on each task separately, while "multi-task" indicates models trained jointly across all selected benchmarks. "⋆" denotes the model is capable of the task, though no number is reported. "-" means incapability in that specific task. "†" indicates that the generalist model embedded previous task-specific models to enhance performance. GiT stands out as the first generalist model to support all listed vision tasks, delivering competitive outcomes without task-specific adaption. Following [15, 42], some generalist models that only report results with task-specific fine-tuning are not included, *e.g.*, OFA [77] and X-Decoder [95]. We highlight the top-1 entries of one-stage multi-task generalist models and joint training improvements with **bold** font. Specific module counts exclude non-computational ones, like index-based text tokenizers.

Methods	Specific Modules		#Params	Object Detection			Instance Seg			Semantic Seg	Captioning		Grounding
	Examples	Num		AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	mIoU(SS)	BLEU-4	CIDEr	Acc@0.5
Specialist Models													
Faster R-CNN-FPN [64]	ResNet,RPN	5	42M	40.3	61.0	44.0	-	-	-	-	-	-	-
DETR-DC5 [12]	ResNet,Encoder	5	41M	43.3	63.1	45.9	-	-	-	-	-	-	-
Deformable-DETR [93]	ResNet,Encoder	5	40M	45.4	64.7	49.0	-	-	-	-	-	-	-
Pix2Seq [19]	ResNet,Encoder	3	37M	43.0	61.0	45.6	-	-	-	-	-	-	-
Mask R-CNN [32]	ResNet,RPN	6	46M	41.0	61.7	44.9	37.1	58.4	40.1	-	-	-	-
Polar Mask [82]	ResNet,FPN	5	55M	-	-	-	30.5	52.0	31.1	-	-	-	-
Mask2Former [23]	ResNet,Decoder	5	44M	-	-	-	43.7	-	-	47.2	-	-	-
VL-T5 [24]	Faster R-CNN	3	440M	-	-	-	-	-	-	-	34.5	116.5	-
UNITER [22]	Faster R-CNN	4	303M	-	-	-	-	-	-	-	-	-	81.4
MDETR [36]	RoBERTa,DETR	6	188M	-	-	-	-	-	-	-	-	-	86.8
Generalist Models (Pre-training + MultiTask-Tuning)													
UniTab [86]	Encoders	4	185M	-	-	-	-	-	-	-	⋆	115.8	88.6
Pix2Seq v2 [20]	ViT,Decoder	2	132M	46.5	⋆	⋆	38.2	⋆	⋆	-	34.9	⋆	⋆
Unified-IO$_{XL}$ [52]	VQ-VAE	4	2.9B	-	-	-	-	-	-	⋆	⋆	122.3	⋆
Shikra-13B [15]	ViT,Vicuna	3	13B	-	-	-	-	-	-	-	⋆	117.5	87.8
Generalist Models (MultiTask-Training)													
Uni-Perceiver [94]	None	1	124M	-	-	-	-	-	-	-	32.0	⋆	⋆
Uni-Perceiver-MoE [92]	None	1	167M	-	-	-	-	-	-	-	33.2	⋆	⋆
Uni-Perceiver-V2 [42]	Mask DINO,Swin	8	308M	58.6†	⋆	⋆	50.6†	⋆	⋆	-	35.4	116.9	⋆
VisionLLM-R50 [78]	Deform-DETR	6	7B	44.6	64.0	48.1	25.1	50.0	22.4	-	31.0	112.5	80.6
GiT-B$_{\text{single-task}}$	None	1	131M	45.1	62.7	49.1	31.4	54.8	31.2	47.7	33.7	107.9	83.3
GiT-B$_{\text{multi-task}}$	None	1	131M	46.7	64.2	50.7	31.9	56.4	31.4	47.8	35.4	112.6	85.8
Improvement (single→multi)				***+1.6***	***+1.5***	***+1.6***	***+0.5***	***+1.6***	***+0.2***	***+0.1***	***+1.7***	***+4.7***	***+2.5***
GiT-L$_{\text{multi-task}}$	None	1	387M	51.3	69.2	55.9	35.1	61.4	34.7	50.6	35.7	116.0	88.4
GiT-H$_{\text{multi-task}}$	None	1	756M	**52.9**	**71.0**	**57.8**	**35.8**	**62.6**	**35.6**	**52.4**	**36.2**	**118.2**	**89.2**

by recent studies. Universal training expands our approach to incorporate 27 comprehensive benchmarks introduced in §4.2. All models leverage AdamW [38] optimizer with a cosine annealing schedule, setting the initial learning rate to 0.0002 and weight decay to 0.05. The largest models of the universal setting are trained on 96 A100 GPUs for 320k iterations. All experiments are evaluated on the selected datasets using standard protocols. More details are in Appendix.

5.2 In-distribution Benchmarking

Comparison with Specialist Models. We compare our single-task model with well-established specialist baselines in Table 4. Our model shows the ability to perform various visual tasks individually within the same framework, narrowing the performance gap with specialized models. It achieves comparable results in most tasks but slightly underperforms in instance segmentation. This is typical for polygon-based methods, which often yield lower results than mask manner. GiT builds a new polygon-based benchmark (+0.9 against PolarMask [82]).

Table 5: Zero shot results. "⋆" and "-" follow Table 4. † are the performance reproduced on mmdetection [14]. "universal" extends multi-task setting by including more datasets.

Methods	Specific Modules Examples	Num	#Params	Object Detection Cityscapes [25]	Instance Seg Cityscapes [25]	Semantic Seg Cityscapes [25]	Semantic Seg SUN RGB-D [67]	Captioning nocaps [2]
Supervised								
Faster R-CNN-FPN [64]	ResNet,RPN	5	42M	40.3	-	-	-	-
Mask R-CNN [32]	ResNet,RPN	6	46M	40.9	36.4	-	-	-
DeepLabV3+ [17]	ResNet,Decoder	3	63M	-	-	80.9	⋆	-
Mask2Former [23]	ResNet,Decoder	5	44M	-	-	80.4	⋆	-
TokenFusion [80]	Segformer,YOLOS	4	-	-	-	⋆	48.1	-
Zero-Shot Transfer								
GLIP-T [45]	Swin,Dy-Head	5	156M	28.1†	-	-	-	-
Grounding DINO-T [49]	Swin,DINO	6	174M	31.5†	-	-	-	-
BLIP-2 (129M) [43]	ViT-G,Qformer	4	12.1B	-	-	-	-	**15.8**
XDecoder-T [95]	FocalNet,Encoder	4	165M	-	16.0	47.3	34.5	⋆
GiT-$B_{\text{multi-task}}$	None	1	131M	21.8	14.3	34.4	30.9	9.2
GiT-$B_{\text{universal}}$	None	1	131M	29.1	17.9	56.2	37.5	10.6
GiT-$L_{\text{universal}}$	None	1	387M	32.3	**20.3**	58.0	39.9	11.6
GiT-$H_{\text{universal}}$	None	1	756M	**34.1**	18.7	**61.8**	**42.5**	12.6

Notably, to maintain a universal interface, our method only uses the simplest label assignments, leaving huge room for performance gains. For example, label assignment used in detection closely mirrors Deformable-DETR [93]. Adopting more advanced strategies like DINO [88] could further improve our results.

Comparison with Generalist Models. Some generalist models [15, 20, 52, 77] employ a two-stage training process, initially leveraging large-scale, task-relevant datasets like image-text pairs or diverse perception data, and then undergoing single- or multi-task downstream tuning within the same framework to enhance performance. Our GiT fully embraces the more challenging one-stage joint training, popularized in LLMs, that blends all data for unified modeling followed by direct downstream evaluation, without any task-specific adaptation.

Table 4 shows that our model not only adeptly manages dense prediction but also outperforms the former leading generalist model, VisionLLM [78], across all tasks, with 50× fewer parameters and a much simpler framework.

Table 4,5,6 show that scaling our model greatly improves multitask, zero- and few-shot performance, sometimes even matching supervised approaches.

Discussion about multi-task capacity. Table 4 reveals that GiT-$B_{\text{multi-task}}$ outperforms GiT-$B_{\text{single-task}}$, showing notable improvements in each task after joint training on five standard datasets. As observed in Table 3, multi-task training typically boosts performance when tasks share the same capabilities but are less effective otherwise. It's clearly observed in the shared localization ability across detection, grounding, and instance segmentation. Conversely, specialized skills, like dense prediction in semantic segmentation and polygon-based regression in instance segmentation don't see significant gains from multi-tasking.

5.3 Out-of-distribution Analysis

Zero-Shot Transfer. After large-scale multi-task training, GiT is readily assessed on a variety of novel data sources. To demonstrate this capability, we conducted zero-shot evaluations on three established datasets across five config-

Table 6: Few shot results of out-distributed domains. We conduct this experiment based on weights pretrained in the universal stage. "⋆" , "-" and † follow Table 5.

Methods	Specific Modules		Medical Imaging@mDice	Remote Sensing@mIoU		Human Centric@mAP	
	Examples	Num	DRIVE [68]	LoveDA [76]	Potsdam [35]	WIDERFace [85]	DeepFashion [50]
Supervised							
U-Net [65]	None	1	81.4	⋆	⋆	-	-
AerialFormer [84]	Encoder,Stem	3	-	54.1	89.1	-	-
RetinaFace [27]	ResNet,FPN	5	-	-	-	52.3	-
Mask R-CNN [32]	ResNet,RPN	6	-	-	-	⋆	59.9
Few-Shot Transfer							
Faster RCNN [64]	ResNet,RPN	4	-	-	-	25.4†	14.9†
DeepLabV3 [16]	ResNet,ASPP	3	32.1†	20.3†	24.2†	-	-
GiT-$B_{\text{multi-task}}$	None	1	34.3	24.9	19.1	17.4	23.0
GiT-$B_{\text{universal}}$	None	1	51.1	30.8	30.6	31.2	38.3
GiT-$L_{\text{universal}}$	None	1	55.4	34.1	37.2	33.4	49.3
GiT-$H_{\text{universal}}$	None	1	**57.9**	**35.1**	**43.4**	**34.0**	**52.2**

urations, addressing four vision tasks beyond visual grounding. These evaluations span a range of contexts, from indoor environments like SUN RGB-D [67], outdoor scenes such as Cityscapes [25], and daily life like nocaps [2]. We report mIoU and SPICE [5] for semantic segmentation and captioning, mAP for object detection and instance segmentation.

As shown in Table 5, our universal models achieve the best results in nearly all tasks. With comparable parameters, GiT-$B_{\text{universal}}$ surpasses X-Decoder [95] on Cityscapes (+8.9) and SUN RGB-D (+3.0) on semantic segmentation, and shows similar advantages in instance segmentation and object detection. Scaling the model further enhances its zero-shot capabilities, nearing supervised performance. BLIP-2 [43] outperforms GiT-H on nocaps, likely attributed to its integration with pretrained language models and extensive training data (129M). Notably, to our knowledge, GiT is the first generalist model to achieve zero-shot performance across various domains and tasks.

Few-Shot Transfer. GiT demonstrates rapid adaptation to out-of-distribution data sources. We conducted a comprehensive few-shot evaluation on five datasets in medical imaging (*i.e.*, DRIVE [68]), remote sensing (*i.e.*,LoveDA [76] and ISPRS [35]), and human-centric scenarios (*i.e.*, WIDERFace [85] and DeepFashion [50]). Our approach follows the N-way K-shot [29] setting (*i.e.*, K=5) and directly fine-tune the pre-trained model on support sets [10].

As for segmentation, we choose DeeplabV3 as baseline, which aligns with the dataset (*i.e.*, ADE20K) used for training our multi-task variant. We observed that both GiT $_{\text{multi-task}}$ and DeeplabV3 perform poorly in few-shot setting. However, after large-scale universal training, GiT-$B_{\text{universal}}$ demonstrates significant improvement. This trend is mirrored in detection, underscoring the advantages of our universal framework for enhancing generalization capabilities.

5.4 Ablation Study

Decoder-only Architecture. Our model follows the GPT's decoder-only design, though its advantages over encoder-decoder frameworks are not well-explored.

Table 7: Ablation of modality experts and text conditioning on GiT-$B_{\text{multi-task}}$, using multiple FFN for multimodal learning and image-to-text attention in visual grounding.

Modality Experts	Text Conditioning	Detection@AP	Ins Seg@AP	Sem Seg@mIoU(SS)	Caption@CIDEr	Grounding@Acc(0.5)
		46.1	31.4	47.8	111.8	78.6
✓		46.2	31.6	47.7	112.2	78.7
	✓	46.7	31.9	47.8	112.6	85.8

Table 8: Ablation study between encoder-decoder and decoder-only architecture.

Methods	Enc Layer	Dec Layer	Detection@AP	Ins Seg@AP	Sem Seg@mIoU(SS)	Caption@CIDEr	Grounding @Acc(0.5)
GiT-$B_{\text{multi-task}}$	12	6	46.3	31.6	46.9	110.8	84.8
GiT-$B_{\text{multi-task}}$	0	18	46.7	31.9	47.8	112.6	85.8

We transformed GiT-B's initial 12 layers into an encoder for image and text, excluding target tokens. Table 8 shows that the encoder-decoder paradigm underperforms decoder-only models. This might be due to decoder-only models allocating more layers (18 vs 6) for processing target tokens.

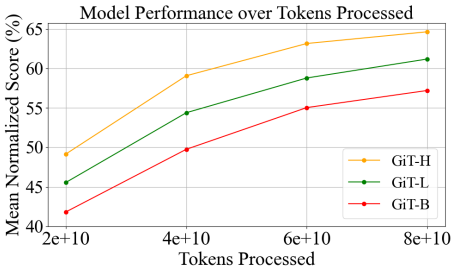

Fig. 5: Model size scaling law results.

Modality Experts. Although employing multiple FFN as modality experts is a commonly used practice [6, 92] for multimodal processing, Table 7 shows no notable performance gains in our approach, leading us to exclude it for achieving a simpler framework.

Text Conditioning. As for visual grounding, we enable image-to-text attention. Table 7 shows the remarkable multi-task improvements, likely due to better differentiation between detection and visual grounding. These two tasks function at distinct image scales (*i.e.*, 1120 and 224), where the former involves identifying multiple boxes and the latter needs to generate a single box guided by text.

Scaling Law Analysis. Figure 5 presents an in-distribution performance of our universal model against its parameter count, offering insights into the potential enhancements with expanding model capacity. We plot performance progression for three model sizes based on a composite score averaging key metrics from all tasks, showing gains with increased scale at a consistent token count.

6 Conclusion

In this paper, we introduce GiT, a simple yet powerful vision foundation model that utilizes only a ViT to integrate diverse visual tasks via a universal language interface. Mirroring multi-task abilities in LLMs, GiT sets new benchmarks in generalist performance. With training over 27 datasets, GiT becomes the first general vision model to excel in zero-shot tasks across diverse domains using shared parameters, showcasing the foundational role of multi-layer transformer.

Acknowledgement

Haiyang Wang would like to thank Mingxu Tao for helpful discussions about language modeling. We sincerely appreciate all the anonymous reviewers for their valuable suggestions.

References

1. Aakanksha, C., Sharan, N., Jacob, D., Maarten, B., Gaurav, M., Adam, R., Paul, B., Won, C.H., Charles, S., Sebastian, G., et al.: Palm: Scaling language modeling with pathways. JMLR (2023)
2. Agrawal, H., Desai, K., Wang, Y., Chen, X., Jain, R., Johnson, M., Batra, D., Parikh, D., Lee, S., Anderson, P.: Nocaps: Novel object captioning at scale. In: ICCV (2019)
3. Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al.: Flamingo: a visual language model for few-shot learning. In: NeurIPS (2022)
4. Alec, R., Jeffrey, W., Rewon, C., David, L., Dario, A., Ilya, S., et al.: Language models are unsupervised multitask learners. OpenAI blog (2019)
5. Anderson, P., Fernando, B., Johnson, M., Gould, S.: Spice: Semantic propositional image caption evaluation. In: ECCV (2016)
6. Bao, H., Wang, W., Dong, L., Liu, Q., Mohammed, O.K., Aggarwal, K., Som, S., Piao, S., Wei, F.: Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. In: NeurIPS (2022)
7. Bavishi, R., Elsen, E., Hawthorne, C., Nye, M., Odena, A., Somani, A., Taşırlar, S.: Introducing our multimodal models (2023), `https://www.adept.ai/blog/fuyu-8b`
8. Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., et al.: On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 (2021)
9. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: NeurIPS (2020)
10. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: CVPR (2017)
11. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: CVPR (2018)
12. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)
13. Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In: CVPR (2021)
14. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 (2019)
15. Chen, K., Zhang, Z., Zeng, W., Zhang, R., Zhu, F., Zhao, R.: Shikra: Unleashing multimodal llm's referential dialogue magic. arXiv preprint arXiv:2306.15195 (2023)

16. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. CVPR (2017)
17. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: ECCV (2018)
18. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., Sutskever, I.: Generative pretraining from pixels. In: ICML (2020)
19. Chen, T., Saxena, S., Li, L., Fleet, D.J., Hinton, G.: Pix2seq: A language modeling framework for object detection. In: ICLR (2022)
20. Chen, T., Saxena, S., Li, L., Lin, T.Y., Fleet, D.J., Hinton, G.E.: A unified sequence interface for vision tasks. NeurIPS (2022)
21. Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft coco captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)
22. Chen, Y.C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Universal image-text representation learning. In: ECCV (2020)
23. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: CVPR (2022)
24. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. In: ICML (2021)
25. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016)
26. Dai, W., Li, J., Li, D., Tiong, A.M.H., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning. NeurIPS (2023)
27. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S.: Retinaface: Single-shot multi-level face localisation in the wild. In: CVPR (2020)
28. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
29. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML. PMLR (2017)
30. Girshick, R.: Fast r-cnn. In: ICCV (2015)
31. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: CVPR (2019)
32. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
33. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
34. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
35. III/4, I.W.: ISPRS 2D Semantic Labeling Contest, `https://www.isprs.org/education/benchmarks/UrbanSemLab/2d-sem-label-potsdam.aspx`
36. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: Mdetr-modulated detection for end-to-end multi-modal understanding. In: ICCV (2021)
37. Kenton, J.D.M.W.C., Toutanova, L.K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (2019)
38. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2015)
39. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)

40. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. IJCV (2017)
41. Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. IJCV (2020)
42. Li, H., Zhu, J., Jiang, X., Zhu, X., Li, H., Yuan, C., Wang, X., Qiao, Y., Wang, X., Wang, W., et al.: Uni-perceiver v2: A generalist model for large-scale vision and vision-language tasks. In: CVPR (2023)
43. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. ICML (2023)
44. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML (2022)
45. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: CVPR (2022)
46. Li, Y., Mao, H., Girshick, R., He, K.: Exploring plain vision transformer backbones for object detection. In: ECCV (2022)
47. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
48. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. NeurIPS (2023)
49. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)
50. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In: CVPR (2016)
51. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
52. Lu, J., Clark, C., Zellers, R., Mottaghi, R., Kembhavi, A.: UNIFIED-IO: A unified model for vision, language, and multi-modal tasks. In: ICLR (2023)
53. Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A.L., Murphy, K.: Generation and comprehension of unambiguous object descriptions. In: CVPR (2016)
54. Ning, J., Li, C., Zhang, Z., Wang, C., Geng, Z., Dai, Q., He, K., Hu, H.: All in tokens: Unifying output space of visual tasks via soft token. In: ICCV (2023)
55. OpenAI: Chatgpt (2022), `https://openai.com/blog/chatgpt`
56. OpenAI: Gpt-4 technical report (2023)
57. Ordonez, V., Kulkarni, G., Berg, T.: Im2text: Describing images using 1 million captioned photographs. NeurIPS **24** (2011)
58. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. NeurIPS **35** (2022)
59. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: ICCV (2015)
60. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
61. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. JMLR (2020)

62. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: ICML (2021)
63. Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S.G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J.T., et al.: A generalist agent. TMLR (2022)
64. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. NeurIPS (2015)
65. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015)
66. Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., Sun, J.: Objects365: A large-scale, high-quality dataset for object detection. In: ICCV (2019)
67. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: CVPR (2015)
68. Staal, J., Abràmoff, M.D., Niemeijer, M., Viergever, M.A., Van Ginneken, B.: Ridge-based vessel segmentation in color images of the retina. TMI (2004)
69. Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., Hashimoto, T.B.: Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca` (2023)
70. Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
71. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
72. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. In: NeurIPS (2017)
73. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
74. Wang, H., Shi, C., Shi, S., Lei, M., Wang, S., He, D., Schiele, B., Wang, L.: Dsvt: Dynamic sparse voxel transformer with rotated sets. In: CVPR (2023)
75. Wang, H., Tang, H., Shi, S., Li, A., Li, Z., Schiele, B., Wang, L.: Unitr: A unified and efficient multi-modal transformer for bird's-eye-view representation. In: ICCV (2023)
76. Wang, J., Zheng, Z., Ma, A., Lu, X., Zhong, Y.: Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. In: NeurIPS (2021)
77. Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., Ma, J., Zhou, C., Zhou, J., Yang, H.: Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In: ICML (2022)
78. Wang, W., Chen, Z., Chen, X., Wu, J., Zhu, X., Zeng, G., Luo, P., Lu, T., Zhou, J., Qiao, Y., et al.: Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. NeurIPS (2023)
79. Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O.K., Singhal, S., Som, S., et al.: Image as a foreign language: Beit pretraining for all vision and vision-language tasks. CVPR (2023)
80. Wang, Y., Chen, X., Cao, L., Huang, W., Sun, F., Wang, Y.: Multimodal token fusion for vision transformers. In: CVPR (2022)
81. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al.: Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 (2016)

82. Xie, E., Sun, P., Song, X., Wang, W., Liu, X., Liang, D., Shen, C., Luo, P.: Polarmask: Single shot instance segmentation with polar representation. In: CVPR (2020)
83. Xu, W., Wang, H., Qi, F., Lu, C.: Explicit shape encoding for real-time instance segmentation. In: ICCV (2019)
84. Yamazaki, K., Hanyu, T., Tran, M., Garcia, A., Tran, A., McCann, R., Liao, H., Rainwater, C., Adkins, M., Molthan, A., et al.: Aerialformer: Multi-resolution transformer for aerial image segmentation. arXiv preprint arXiv:2306.06842 (2023)
85. Yang, S., Luo, P., Loy, C.C., Tang, X.: Wider face: A face detection benchmark. In: CVPR (2016)
86. Yang, Z., Gan, Z., Wang, J., Hu, X., Ahmed, F., Liu, Z., Lu, Y., Wang, L.: Unitab: Unifying text and box outputs for grounded vision-language modeling. In: ECCV (2022)
87. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: ECCV. Springer (2016)
88. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In: ICLR (2022)
89. Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X.V., et al.: Opt: Open pre-trained transformer language models. arXiv preprint arXiv:2205.01068 (2022)
90. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: CVPR (2017)
91. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023)
92. Zhu, J., Zhu, X., Wang, W., Wang, X., Li, H., Wang, X., Dai, J.: Uni-perceiver-moe: Learning sparse generalist models with conditional moes. NeurIPS (2022)
93. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. ICLR (2020)
94. Zhu, X., Zhu, J., Li, H., Wu, X., Li, H., Wang, X., Dai, J.: Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. In: CVPR (2022)
95. Zou, X., Dou, Z.Y., Yang, J., Gan, Z., Li, L., Li, C., Dai, X., Behl, H., Wang, J., Yuan, L., et al.: Generalized decoding for pixel, image, and language. In: CVPR (2023)