Supplementary to 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

In our supplementary, we provide detailed information including model design specifics in §1, dataset summaries in §2, along with in-depth training, inference and evaluation procedures in §3 and 4. Additional ablation experiments are included in §5. §6 details specific modules used in comparative methods. Qualitative results across different datasets and tasks are in §7. Lastly, limitations, negative societal impacts, and a comparison with Fuyu-8B are in §8.

1 Implementation details

Window Attention. Our window attention is adapted from the SAM [28] variant of ViT [18]. Following SAM, after patch embedding, images are downsampled by a factor of 16, and windows are defined with a size of 14×14. The primary distinction from the original lies in how we handle multi-track local observations and responses in the parallel training stage, such as grid-wise prompts (*i.e.*, local image token, task identifier) and their outputs. To manage these multi-track elements, we merge them into a sequence and append them after the shared observation. Consequently, the input to window attention consists of multiple parts, requiring a customized attention mask to ensure grid independence while enabling autoregressive prediction, as detailed in Figure 1. Within each subprocess group (*i.e.*, those associated with the same grid), interactions are left-to-right unidirectional attention. Moreover, tokens belonging to different subprocesses are isolated, preventing them from accessing each other's information.

Global Attention. In tasks that require object- and pixel-level analysis, the large number of local predictions creates significant memory and computational burdens, especially in global attention layers, where processing attention across all grid points can be unnecessary and inefficient. Therefore, for such tasks, we have optimized the global attention layer to focus only on the shared global observations (*i.e.*, input image and text), eliminating the need to compute targets for each grid. Table 1 shows that this strategy slightly impacts performance but greatly decreases computation time. However, in captioning and visual grounding with a 224 image size, which involves only one window and a single global response, this optimization is unnecessary.

Table 1: Performance of semantic segmentation by single-task training with our accelerated global attention. It significantly reduces the computational cost with slight performance drops.

Global Attention	mIoU	Training Time
Normal	47.9	51h
Accelerated	47.7	35h

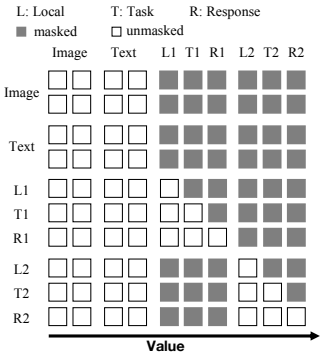

Fig. 1: Attention mask visualization.

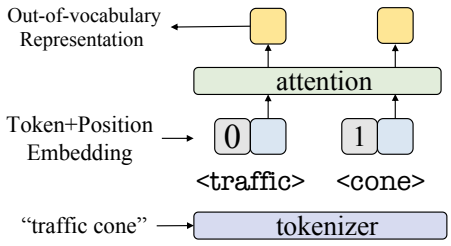

Fig. 2: Out-of-vocabulary representation.

Out-of-vocabulary Representation. We encode multi-piece out-of-vocabulary concepts to a single token. This is achieved through a streamlined approach that utilizes only one attention layer combined with absolute positional encoding. As shown in Figure 2, "traffic cone" is tokenized as <traffic><cone>. The corresponding text embeddings, augmented with positional encoding, are input into the attention layer, allowing each word to interact with the rest. We select the first output token as the final representation for multi-word concepts like "traffic cone". For single-word concepts, we use the original text embedding directly.

Background Representation. Given that each dataset contains distinct positive and negative classes, utilizing text labels like <background> to denote negative classes could lead to ambiguity when training across multiple datasets. Therefore, we employed a unique encoding approach for the background class,

$$\mathcal{F}_{\text{background}} = -\sum_{i=0}^{N-1} \mathcal{F}_i / N \tag{1}$$

where $\mathcal{F}_i$ is the representation of i-th positive class and N denotes the total number of categories. This approach makes the cosine similarity between tokens of a positive class and those assigned to the background class typically negative. Its superior performance in zero-shot scenarios highlights its effectiveness.

Resolution and Coordinate Discretization. For our experiments, we use different image resolutions tailored to specific tasks: 1120 × 1120 pixels for object

detection and instance segmentation, 672 × 672 pixels for semantic segmentation, and 224 × 224 pixels for image captioning and visual grounding. To encode spatial positions as discrete tokens, we discretize the image coordinates into a set number of intervals. Specifically, we determine the number of these intervals to be double the resolution of the input image. For instance, with an input image of 224 × 224 pixels, we divide the coordinate space into 448 discrete intervals.

2 Extended Datasets

2.1 In-distribution Datasets

During universal training, a total of 27 datasets from 16 publicly accessible data sources are used, with sizes and weights detailed in Table 2. Note that the actual quantities in web-sourced caption datasets (CC3M [48], CC12M [6], SBU Captions [43]) are fewer than the original number reported due to inactive links.
COCO. The MS COCO dataset, or Microsoft Common Objects in Context [36], is a comprehensive dataset for object detection, segmentation, key-point detection, and captioning. It includes over 330K images, with annotations for more than 220K, featuring 1.5 million objects across 80 categories. Each image has five sentence descriptions and 250K pedestrians are annotated with keypoints. The initial release in 2014 has 164K images in training (83K), validation (41K), and test (41K) sets. In 2017, the training/validation split changed to 118K/5K.
Objects365. Objects365 [47] is a vast object detection dataset, comprising 365 object categories and boasting over 2 million training images along with 30 million annotated bounding boxes. This dataset presents diverse objects in different scenarios, providing a robust benchmark for challenging object detection tasks.
OpenImages. Open Images [31] is a dataset with about 9 million images, each annotated with image-level labels, object bounding boxes, segmentation masks, visual relationships, localized narratives, and point-level labels. Covering 20,638 image-level labels, 600 object classes with 16 million bounding boxes, and 2.8 million segmentation masks, it stands as a valuable resource in computer vision.
LVIS. LVIS [22] (Large Vocabulary Instance Segmentation) is a dataset tailored for instance segmentation tasks, providing approximately 2 million high-quality segmentation masks across over 1000 entry-level object categories within a dataset of 164,000 images. This dataset was created to tackle the Zipf distribution commonly observed in natural images, making it an invaluable resource for researchers and developers working on instance segmentation tasks dealing with a large vocabulary of objects.
Pascal VOC 2007. The Pascal VOC 2007 [19] dataset serves as a crucial resource for real-world object recognition, featuring 20 object classes. With 9,963 photos and 24,640 labeled samples, thoughtfully split for balanced training/validation and testing, it stands as a versatile dataset supporting various tasks, including classification, detection, segmentation, and person layout.
Pascal VOC 2012. Pascal VOC 2012 [19] is a valuable dataset for recognizing objects in real-world settings. It encompasses 20 object classes and includes 11,530 images with 27,450 ROI-tagged objects and 6,929 segmentations, serving

Table 2: Universal training dataset details. Columns from left to right indicate dataset size, proportion to total data, assigned group number, and sampling weight. Weights are evenly distributed across the tasks. Different scenarios within each task (*e.g.*, daily life, autonomous driving) create individual groups with equal weights. Sampling weights in groups are set based on dataset sizes.

Dataset	Size	Percent (%)	Group ID	Weight (%)
Object Detection	3.8M	22.55	-	20.00
Objects365 [47]	1.7M	9.98	0	3.22
OpenImages [31]	1.7M	9.98	0	3.22
LVIS [22]	164K	0.96	0	0.23
nuImages [3]	93K	0.55	1	6.66
Pascal VOC 2007 [19]	10K	0.06	2	0.37
Pascal VOC 2012 [19]	11K	0.06	2	0.22
COCO 2017 [36]	164K	0.96	2	6.07
Instance Segmentation	1.4M	8.34	-	20.00
LVIS [22]	164K	0.96	3	0.76
OpenImages [31]	1M	5.87	3	5.90
nuImages [3]	93K	0.55	4	6.66
COCO 2017 [36]	164K	0.96	5	6.66
Semantic Segmentation	322K	1.89	-	20.00
COCO-Stuff [4]	164K	0.96	6	6.28
Pascal Context [41]	10K	0.06	6	0.38
nuImages [3]	93K	0.55	7	4.84
BDD100K [60]	10K	0.06	7	0.52
Mapillary Vistas [42]	25K	0.15	7	1.30
ADE20K [61]	20K	0.12	8	6.67
Image Caption	11.3M	66.54	-	20.00
CC3M [48]	1.8M	10.57	9	1.74
CC12M [6]	7.8M	45.79	9	6.96
SBU Captions [43]	800K	4.70	9	0.58
Visual Genome [29]	770K	4.52	9	0.71
COCO Caption [12]	164K	0.96	10	10.00
Visual Grounding	115K	0.68	-	20.00
RefCOCO [27]	20K	0.12	11	4.00
RefCOCO+ [27]	20K	0.12	11	4.00
RefCOCOg [40]	25K	0.15	11	4.00
RefCLEF [27]	20K	0.12	12	4.00
Flickr30K [44]	30K	0.18	13	4.00
All	17M	100	-	100

as a prominent benchmark in computer vision.

nuImages. The nuImages [3] dataset complements the nuScenes [3] for autonomous driving by providing 93,000 2D annotated images, with 1.2 million camera images from past and future timestamps. It is part of the nuScenes ecosystem and focuses on panoptic and multi-annotation aspects. The dataset covers various driving scenarios, including diverse conditions such as rain, snow, and night. It also offers temporal dynamics with 2 Hz spaced images. The annotations encompass 800,000 foreground objects with instance masks and 100,000 semantic segmentation masks.

ADE20K. The ADE20K [61] semantic segmentation dataset comprises 20,000 scene-centric images meticulously annotated at the pixel level for both objects and object parts. Encompassing 150 semantic categories, it includes items like sky, road, and specific objects such as person, car, and bed. The dataset is divided into 20,210 training, 2,000 validation, and 3,000 testing images.

COCO-Stuff. The COCO-stuff [4] dataset holds significance for diverse scene understanding tasks, such as semantic segmentation, object detection, and image captioning. Derived by augmenting the original COCO dataset, which initially prioritized object annotations, it addresses the oversight of stuff annotations. Spanning 164,000 images, the COCO-stuff dataset includes 172 categories, incorporating 80 things, 91 stuff, and 1 unlabeled class.

Pascal Context. The PASCAL Context [41] dataset extends the PASCAL VOC 2010 [19] detection challenge by providing pixel-wise labels for all training images. Encompassing over 400 classes, which include the original 20 classes from PASCAL VOC segmentation, these classes are categorized into objects, stuff, and hybrids. To address the sparsity of many object categories, a common practice involves using a subset of 59 frequently occurring classes.

BDD100K. BDD100K [60] is a large dataset with 100K videos, providing over 1,000 hours of driving experience and 100 million frames. It includes annotations for road objects, lane markings, drivable areas, and detailed instance segmentation. For road object detection and drivable area segmentation challenges, there are 70,000 training and 10,000 validation images. For full-frame semantic segmentation, there are 7,000 training and 1,000 validation images.

Mapillary Vistas. Mapillary Vistas [42] is a large-scale street-level image dataset with 25,000 high-resolution images. Featuring annotations for 66 object categories, including instance-specific labels for 37 classes, it adopts a dense and fine-grained annotation style using polygons. The dataset primarily focuses on semantic image segmentation and instance-specific image segmentation, aiming to advance visual road-scene understanding.

CC3M. Conceptual Captions, known as CC3M [48], features an extensive collection of around 3.3 million images, each meticulously paired with descriptive captions. Extracted from Alt-text HTML attributes associated with web images, these captions undergo an automated pipeline for quality assurance. This makes the dataset highly versatile, catering to a diverse range of natural language processing and image understanding tasks.

CC12M. Conceptual 12M [6] (CC12M) is a dataset specifically created for

vision-and-language pre-training. It consists of a substantial 12 million image-text pairs. Unlike some other datasets with restrictive requirements, CC12M relaxes its data collection pipeline to enhance dataset scale and diversity. It has been shown to provide state-of-the-art results in vision-and-language tasks, particularly in long-tail visual recognition, making it a valuable resource for research and development in this field.

SBU Captions. The SBU Captions dataset [43] is a collection of 1 million images and their associated captions sourced from Flickr, primarily used for training image captioning models. It provides diverse real-world images and textual descriptions, serving as a valuable resource for research in computer vision and natural language processing.

Visual Genome. Visual Genome [29] is a comprehensive dataset with 108,077 images, richly annotated with 5.4 million region descriptions, 1.7 million visual question answers, 3.8 million object instances, 2.8 million attributes, and 2.3 million relationships. This dataset is designed to provide detailed information about images, including objects, attributes, and the relationships between them.

COCO Caption. COCO Captions [12] consists of 1.5 million captions for 330,000 images, with five captions for each image in the training and validation sets. The "Karpathy split", a widely used subset of this dataset created by Andrej Karpathy, involves merging the train and val sets from the raw dataset, creating a new validation set by selecting 5,000 images from the original val set, and an additional 5,000 images are used to form a test set.

RefCOCO. The RefCOCO [27], RefCOCO+ [27], and RefCOCOg [40] datasets were generated through the ReferitGame, a two-player game where one participant describes a segmented object in an image using natural language, and the other participant identifies the correct object. In RefCOCO, there are no language restrictions on referring expressions, whereas in RefCOCO+, location words are prohibited. These datasets concentrate on appearance-based descriptions, such as "the man in the yellow polka-dotted shirt," rather than perspective-dependent ones. RefCOCO comprises 142,209 referring expressions for 50,000 objects in 19,994 images, and RefCOCO+ contains 141,564 expressions for 49,856 objects in 19,992 images.

RefCLEF. RefCLEF [27], also known as ReferIt, consists of 20,000 images sourced from the IAPR TC-12 dataset, accompanied by segmented image regions from the SAIAPR-12 dataset. The dataset is evenly split into two sections: one with 10,000 images designated for training and validation, and another with 10,000 images for testing. The training and validation portion includes a total of 59,976 entries, each consisting of an image, a bounding box, and a description. Test set is slightly larger, featuring 60,105 entries with the same type of data.

Flickr30K. Flickr30K [44] is a widely recognized dataset used for sentence-based image descriptions. It features 31,783 images depicting everyday activities and events, each accompanied by a descriptive caption. This dataset serves as a standard benchmark for studying the relationship between linguistic expressions and visual media.

2.2 Out-distribution Datasets

Cityscapes. Cityscapes [16] is a large dataset for understanding urban scenes, featuring semantic, instance-wise, and pixel-level annotations across 30 classes grouped into 8 categories. It comprises around 5,000 finely annotated images and 20,000 coarsely annotated ones, recorded in various cities under different conditions. This dataset is valuable for tasks related to urban scene analysis.
SUN RGB-D. The SUN RGB-D dataset [50] comprises 10,335 RGB-D images of room scenes, each with depth and segmentation maps. It's annotated for 700 object categories and divided into training and testing sets with 5,285 and 5,050 images, respectively. This dataset addresses the need for large-scale 3D annotations and metrics for scene understanding tasks. It includes data from four sensors, with extensive annotations for 2D and 3D object boundaries, orientations, room layout, and scene categories, enabling advanced algorithm training and cross-sensor bias study.
nocaps. The nocaps [1] dataset pushes image captioning models to grasp a wider array of visual concepts from diverse data origins. Comprising 166,100 human-generated captions for 15,100 images sourced from OpenImages, the dataset integrates different training data, including COCO image-caption pairs and OpenImages labels and bounding boxes, with a specific emphasis on describing objects.
DRIVE. The DRIVE [51] dataset used for retinal vessel segmentation consists of 40 color fundus images, including 7 displaying abnormal pathology. Captured during diabetic retinopathy screenings in the Netherlands, these images were taken with a Canon CR5 camera featuring a 45-degree field of view. The dataset is split into a training set (20 images) and a testing set (20 images), each accompanied by a circular field of view (FOV) mask. Expert manual segmentations are provided for assessment in the training set, while the testing set includes two observer-based segmentations, with the first observer's results considered as the ground truth for evaluation.
LoveDA. The LoveDA [52] dataset comprises 5987 high-resolution remote sensing images (0.3 m) from urban and rural areas in Nanjing, Changzhou, and Wuhan. It targets semantic segmentation and domain adaptation tasks, offering challenges such as multi-scale objects, complex backgrounds, and inconsistent class distributions, aiming to address diverse geographical environments.
ISPRS Potsdam. The ISPRS Potsdam [24] dataset comprises 38 patches with true orthophotos (TOP) and digital surface models (DSM) having a 5 cm ground sampling distance. The TOP images are available in various channel compositions (IRRG, RGB, RGBIR), and DSM files contain 32-bit float values representing heights. Some patches have normalized DSMs, indicating heights above the terrain. Ground truth labels are provided for a portion of the data, with the rest reserved for benchmark testing.
WIDER Face. The WIDER Face [57] dataset is a comprehensive face detection benchmark dataset, consisting of 32,203 images with a diverse range of 393,703 labeled faces. These images exhibit variations in scale, pose, and occlusion. The dataset is categorized into 61 event classes, with 40% for training, 10% for validation, and 50% for testing. Evaluation follows the PASCAL VOC dataset metric.

DeepFashion. The DeepFashion [38] dataset is a comprehensive collection of around 800,000 fashion images, accompanied by extensive annotations. These annotations include 46 fashion categories, 1,000 descriptive attributes, bounding boxes, and landmark information. The dataset covers a broad spectrum of fashion images, from well-posed product photos to real-world consumer snapshots.

3 Training

3.1 Implementation Details

Training schemes. For single-task training, GiT-$B_{\text{single-task}}$ is typically trained using a batch size of 24 for 120,000 iterations on 8 NVIDIA A100 GPUs (40GB), following a cosine annealing schedule. In multi-task joint training on five datasets, GiT-$B_{\text{multi-task}}$ undergoes training with the same batch size and GPU number for more iterations (*i.e.*, 640,000). The large and huge model variants require more GPU memory for training and are therefore trained on 12 and 24 GPUs, respectively. For large-scale universal training, we train all models using a batch size of 96 across 320,000 iterations. This process is conducted on setups of 32, 48, and 96 GPUs, resulting in total training times of 3, 5, and 7 days, respectively.
Custom learning rate. For the layers without pretraining, we applied the standard base learning rate. In contrast, the layers that had been pretrained used progressively increasing learning rates. This strategy begins with a learning rate that is 0.1 times the base rate for the first pretrained layer, gradually escalating to a full 1.0 times the base rate by the final pretrained layer. We argue this method enhances the integration of pretrained and newly trained weights, leading to better overall performance of the model.
Grid generation and sampling. We adjust the grid sizes according to the level of detail required by each task. For object detection and instance segmentation, we work with 5×5 grids in each window, while for semantic segmentation, we increase the grid size to 14×14. To illustrate, in object detection, an input image of 1120×1120 pixels is represented by a 25×25 grids, and in semantic segmentation, a 672×672 pixels is represented by a 42×42 grids. Computing losses for every point on these grids would demand excessive computational resources, particularly for semantic segmentation. To manage this, we employ a strategy of sampling specific grid points during training, selecting a predetermined number of points with a focus on including positive samples and supplementing with negative samples as needed. Specifically, for object detection and instance segmentation, we choose 10 points out of 25 in each window, and for semantic segmentation, we select 32 points out of 196. As shown in Table 3, this method effectively reduces computational costs without significant performance drops.

3.2 Label Assignment

Object Detection. Our approach employs the well-established Hungarian matching algorithm [30] for label assignment calculation. For each grid point, we compute its normalized L1 distance to the centers of all boxes as the matching cost.

Table 3: Performance of grid sampling on object detection with 25 × 25 grid resolution.

Sample Number	mAP	Training Time
625	45.3	47h
250	45.1	20h

Table 4: The evaluation results of the models after universal training on five standard vision-centric benchmarks.

Methods	#Params	Object Detection			Instance Seg			Semantic Seg	Captioning		Grounding
		AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	mIoU(SS)	BLEU-4	CIDEr	Acc@0.5
GiT-$B_{universal}$	131M	44.4	61.2	48.1	30.3	53.0	30.0	44.6	33.6	108.3	84.2
GiT-$L_{universal}$	387M	50.2	67.6	54.6	33.1	58.4	32.7	48.1	36.2	117.5	86.0
GiT-$H_{universal}$	756M	53.3	71.2	58.3	35.9	62.6	36.1	53.0	37.7	124.2	88.3

Instance Segmentation. Similar to object detection, instance segmentation targets are determined by computing the L1 distance between bounding box centers and grid positions. Polar coordinates with 24 rays, inspired by PolarMask [55], are employed for mask representation. The mass center of an object is calculated using its annotated polygon boundaries. Grid points classified as positive must accurately predict object category, bounding box, centroid, and distances from the mass center to boundary points.

Semantic Segmentation. Expanding upon ViT, we generate patch features (42 × 42) by downsampling the image (672 × 672) via a factor of 16. Given the dense prediction nature of semantic segmentation, we align the grid point size with the patch feature size. To alleviate computational load, we downsample original mask annotations (672 × 672) by a factor of 4, resulting in annotations of size 168 × 168, which is four times larger than the grid size. Subsequently, each grid point autonomously predicts segmentation annotations for 16 positions within a 4 × 4 square centered around it.

Image Captioning. In our image captioning process, we tokenize each caption into a fixed-length sequence of 20 tokens. If the caption length is shorter than 20 tokens, we pad it with termination symbols to ensure uniformity.

Visual Grounding. In visual grounding tasks, each query directly targets a specific bounding box, removing the necessity to align boxes with grid points.

3.3 Data Augmentation

Object Detection and Instance Segmentation. For object-level perception tasks, images undergo preprocessing steps. Initially, images are horizontally flipped with a 0.5 probability. Subsequently, two methods are employed to achieve a fixed input size. The first method involves direct resizing of the image to dimensions of 1120 × 1120, disregarding the original aspect ratio. The second method randomly resizes the image to one of three size pairs: (400, 4200), (500, 4200), or (600, 4200), while preserving the original aspect ratio. Following resizing, the image is cropped to a size of (384, 600) and then resized again to

Table 5: Universal training evaluation results on detection, instance segmentation, and visual grounding datasets.

Methods	Object Detection@AP					Grounding@Acc				Instance Seg@AP
	Objects365 [47]	OpenImages [31]	LVIS [22]	VOC0712 [19]	nuImages [3]	RefCOCO+ [27]	RefCOCOg [40]	Flickr30K [44]	RefCLEF [27]	LVIS [22]
GiT-$B_{universal}$	17.7	43.4	12.3	79.0	44.5	72.5	76.9	71.0	72.2	8.4
GiT-$L_{universal}$	25.5	51.6	17.3	83.6	47.2	73.9	78.9	72.7	74.5	11.4
GiT-$H_{universal}$	31.9	57.7	21.7	84.9	50.0	78.3	80.7	77.5	75.8	14.8

Table 6: Evaluation of universal training on segmentation datasets, with all results measured using the mIoU metric.

Methods	COCO-Stuff [4]	Pascal Context [41]	BDD100K [60]	Mapillary Vistas [42]
GiT-$B_{universal}$	42.6	56.8	57.8	23.0
GiT-$L_{universal}$	46.0	60.4	59.3	25.4
GiT-$H_{universal}$	49.1	63.3	61.5	28.9

1120 × 1120 pixels.

Semantic Segmentation. In semantic segmentation, specific preprocessing steps are applied to images to ensure their size is standardized and to increase diversity. Initially, images are acquired with a size of 672 × 672 pixels, employing random selection between two methods. The first method directly resizes the image to 672 × 672, disregarding the original aspect ratio. The second method involves scaling the image to sizes ranging from 100% to 200% of 672, again without preserving the original aspect ratio. Following this, a random crop is applied to ensure the image size remains 672 × 672 pixels. Moreover, to augment image diversity, two additional operations are performed with a 50% probability: horizontal flipping and photometric distortions. These steps collectively contribute to a more robust dataset for segmentation tasks.

Image Captioning. As for this task, we initiate preprocessing with a dynamic crop, varying size ratio in [0.08, 1.0] and aspect ratio in [3/4, 4/3] in relation to the original image. Following this crop, the image is resized to 224×224 dimensions. Additionally, there is a 50% probability of horizontally flipping the image for further augmentation.

Visual Grounding. Visual grounding augmentation includes color adjustments with a 50% probability, enabling changes in brightness, contrast, saturation, and hue. Subsequently, the image undergoes a random crop within a relative range of (0.8, 0.8) of the original size. Finally, we resize the image to 224×224 without keeping the original aspect ratio.

4 Evaluation

4.1 Auto-regressive Decoding

We tailor unique decoding rules for various tasks based on task templates. For example, in object detection, using the template $<\texttt{c}><\texttt{x}_1><\texttt{y}_1><\texttt{x}_2><\texttt{y}_2>$, the category is decoded in the first position, drawing from a vocabulary containing all categories in the dataset. The subsequent four positions decode numerical

Table 7: Decoding steps for all five tasks.

Task	Object Detection	Instance Segmentation	Semantic Segmentation	Image Captioning	Visual Grounding
Decoding Step	5	31	16	20	4

Table 8: Inference speed of GiT-B on A100.

Task	Resolution	Grid Number	Decoding Step	FPS
Object Detection	1120 × 1120	625	5	2.5
Instance Segmentation	1120 × 1120	625	31	0.7
Semantic Segmentation	672 × 672	1764	16	1.5
Image Captioning	224 × 224	1	20	3.2
Visual Grounding	224 × 224	1	4	8.1

Table 9: Latency comparison with SAM on semantic segmentation task.

Method (ADE20K)	Resolution	#Params	FPS
SAM-B [41]	672 × 672	90M	1.6
GiT-B	672 × 672	131M	1.5

values, drawing from a vocabulary of discretized locations. Table 7 illustrates the fixed decoding step number for all tasks, with no terminator token required except for image captioning. In image captioning, predictions following the terminator are disregarded during inference.

4.2 Inference Speed

In Table 8, we present the inference speed of GiT-B across five tasks, measured on a single NVIDIA A100 GPU with a batch size of 1. Due to our adherence to the auto-regressive decoding paradigm commonly seen in NLP, we inherit the drawback of slow inference speed. This limitation becomes more pronounced in high-resolution object-level and semantic segmentation tasks that necessitate per-pixel predictions. However, we contend that leveraging multiple parallel decoding has significantly improved our method's speed, bringing it to an acceptable level. As shown in Table 9, our approach demonstrates comparable segmentation speed to SAM. Given that our structure and prediction approach closely align with LLM, the inference acceleration techniques [7] employed for LLM also hold promise for enhancing our method.

4.3 Benchmarking Setup

Multi-Task Learning. On the multi-task datasets, we conducted evaluations on the validation sets, except for COCO Caption [4], where we used the Karpathy split [26] for evaluation on the test set.
Universal Learning. We evaluate our universal models on several key datasets. Table 4 presents their performance on representative datasets for five tasks. However, due to the less frequent sampling of these analyzable multi-task datasets during universal training, their performance slightly lags behind models trained on multi-task benchmark. For further performance insights on other datasets, refer to Tables 5 and 6. Notably, for image captioning, all datasets except COCO Caption are entirely used in training, obviating the need for extra evaluation.
Few-shot Learning. We adopt the classical N-way K-shot [20] setting to create a support set for few-shot evaluation. In this setup, for each class in the dataset,

Table 10: Few shot datasets.

Dataset	Size	Category Number	Support Set Size	Training Iters
DRIVE [51]	40	2	10	100
LoveDA [52]	5,987	7	35	100
ISPRS Potsdam [24]	5,472	6	30	100
WIDERFace [57]	32,203	1	5	50
DeepFashion [38]	800,000	15	75	100

Table 11: Ablation of text conditioning on visual grounding task.

Models	Text Conditioning	Acc@0.5
GiT-B$_{\text{single-task}}$		82.7
GiT-B$_{\text{single-task}}$	✓	83.3
GiT-B$_{\text{multi-task}}$		78.6
GiT-B$_{\text{multi-task}}$	✓	85.8

Table 12: Ablation of beam number on image captioning task.

Beam Number	BLEU-4	CIDEr
1	33.1	106.9
2	33.5	107.2
3	33.7	107.9
5	33.7	107.6

we extract k samples labeled with the corresponding class, resulting in the selection of N×K samples. By default, K is set to 5. As depicted in Table 10, we sample varying quantities of support sets depending on the number of categories in each dataset. Each experiment, by default, iterates 100 times on the support set. However, due to the limited size of the support set in WIDERFace [57], we reduce the iteration count to 50 times to mitigate the risk of overfitting. All few-shot training is conducted with a fixed learning rate of 2e-4.

We select Faster R-CNN [45] and DeepLabV3 [9], two classic methods, as comparative baselines. In the case of Faster R-CNN, we employ the version with ResNet-50 as the backbone, utilizing pre-trained weights from the COCO [36] dataset. For DeepLabV3, we opt for the version with ResNet-101 as the backbone, leveraging pre-training on the ADE20K [61] dataset.

5 More ablation studies

Text Conditioning. In visual grounding, we incorporate image-to-text attention during network forwarding, enhancing task differentiation between detection and visual grounding. Table 11 demonstrates that incorporating text conditioning results in a modest improvement of +0.6 in visual grounding when trained independently. However, its impact becomes more significant in multi-task training, showing a remarkable enhancement of +7.2, aligning with our hypothesis.
Beam Search. Table 12 demonstrates how performance varies with different beam sizes in beam search. We observe an improvement as the beam size increases from 1 to 2, but performance stabilizes between 2 and 5, with only a minor drop in CIDEr. Given that larger beam sizes lead to longer inference times, we have selected a default beam size of 2.
Mass Center and Ray Number. Table 16 presents an ablation of instance segmentation settings. Utilizing mass center yields better results than box cen-

ter, probably because the box center might fall outside the object. Employing 36 rays slightly improves performance but at the cost of significant training time.

Model initialization. Our model is initialized with SAM encoder. Table 14 indicates that SAM weight excels in fine-grained tasks, while MAE is better for image-level tasks.

Interaction among instances. GiT's flexibility allows us to easily integrate popular instance interaction methods from [5, 10]. Specifically, we apply global self-attention, similar to DETR, across all grid local tokens <Local>. For causal attention, we also use it with these local prompts. Table 13 shows their improvements.

Number of New Layers. Table 15 shows adding just one new layer can significantly boost performance, improving mAP by 2.6, likely due to the difference between image input and language targets. Involving more layers continues to improve results, with gains leveling off after six layers.

Table 13: Interaction.

Attention Type	mAP
no	45.1
causal	45.4
global self	45.5

Table 14: Ablation study on initialization.

Pretrain Weight	Detection AP	Ins Seg AP	Sem Seg mIoU(SS)	Caption CIDEr	Grounding P@0.5
SAM	46.7	31.9	47.8	112.6	85.8
MAE	44.1	29.7	46.2	113.7	87.9

6 Specific Modules of Comparison Methods

In Table 17, we outline the specific modules and parameter quantities utilized for method comparison. Many methods, regardless of whether they are specialist or generalist models, incorporate task-specific modules and modality-specific encoders in their designs. In contrast, our approach is characterized by its simplicity, as it does not rely on such intricate designs.

7 Visualization

Task Visualization. In Figure 4, we visualize an example for each task, showcasing the image input, text-formatted predictions, and the visualization of the prediction results from left to right. For simplicity, we selected a few examples of local responses predicted by the model and listed their corresponding text-formatted predictions.

Zero-shot Visualization. In Figure 5, we showcase qualitative examples of predictions on zero-shot datasets made by GiT-$H_{\text{universal}}$. Notably, our model accurately predicts missing annotations in some cases. For instance, in Cityscapes detection, it correctly identifies unannotated bicycles and vehicles, even under

Table 15: Ablation study of new layer on GiT-$B_{\text{single-task}}$.

New Layers	Detection@AP
0	40.2
1	42.8
2	43.8
3	44.6
6	45.1

Table 16: Ablation on instance segmentation settings.

Box Center	Mass Center	Ray Number	mAP	Training Time
✓		24	29.0	32h
✓		36	29.2	49h
	✓	24	31.4	32h
	✓	36	31.7	49h

low-light conditions. A similar accuracy is observed in SUN RGB-D segmentation, where the model detects all chairs, although only two are annotated. In Cityscapes segmentation, despite the dataset's bias of excluding self-owned vehicles from annotation, our model demonstrates exceptional generalization by correctly classifying these vehicles, relying on minimal information and without dataset-specific fine-tuning.

Few-shot Visualization. Figure 6 provides visual representations of the qualitative predictions made by GiT-$H_{\text{universal}}$ on few-shot datasets. These examples highlight the remarkable performance of our model in situations with limited data, emphasizing its potential for applications across diverse domains.

8 Discussion

Comparison with Fuyu-8B. Compared to Fuyu-8B [2], which focuses on well-explored vision-language tasks, our GiT extends the scope of the multi-layer transformer to often-overlooked object and pixel-level tasks with a universal language interface. To achieve it, we design a flexible parallel decoding template using point prompts for task unification across various perceptual scales. The local image prompt is also introduced to enhance fine-grained perception ability.

Comparison with adapter-based methods. Our method provides an alternative solution for LVMs. Unlike previous fine-tuning efforts with LLMs, we aim to close the architectural gap between vision and language. Moreover, our GiT allows easy end-to-end implementation without module-specific design, greatly simplifying the training process and model scaling.

Limitations. Constrained by training data limited to five selected tasks with relatively straightforward task prompts, GiT struggles to generalize to entirely new tasks in zero-shot settings. Task-level zero-shot remains challenging, even for capable LLMs. GiT closely aligns with it and inherits this limitation. However, our GiT shows strong extendibility in task unification, potentially supporting

Table 17: Specific modules and their corresponding parameter quantities for the methods used for comparison. The parameter of text embedding is excluded because it operates in a zero-computation index manner.

Methods	Specific Modules	Num	#Params
Specialist Models			
Faster R-CNN-FPN [45]	ResNet,FPN,RPN,ClassficationHead,RegressionHead	5	42M
DETR-DC5 [5]	ResNet,Encoder,Decoder,ClassficationHead,RegressionHead	5	41M
Deformable-DETR [63]	ResNet,Encoder,Decoder,ClassficationHead,RegressionHead	5	40M
Mask R-CNN [23]	ResNet,FPN,RPN,RPNHead,ClassficationHead,RegressionHead	6	46M
Polar Mask [55]	ResNet,FPN,ClassficationHead,CenternessHead,RegressionHead	5	55M
Mask2Former [14]	ResNet,PixelDecoder,TransformerDecoder,ClassficationHead,MaskHead	5	44M
Pix2Seq [10]	ResNet,Encoder,Decoder	3	37M
UNITER [13]	Faster R-CNN,Project Layer, Encoder,Decoder	4	303M
VILLA [21]	Faster R-CNN, Encoder,Decoder	3	369M
MDETR [25]	CNN,RoBERTa,Image Adapter, Text Adapter,Encoder,Decoder	6	188M
VL-T5 [15]	Faster R-CNN, Encoder,Decoder	3	440M
DeepLabV3+ [9]	ResNet,Decoder,Auxiliary Head	3	63M
TokenFusion [54]	Segformer,YOLOS,Fusion Module,GroupFree	4	79M
U-Net [46]	Encoder,Decoder,Decode Head	3	8M
AerialFormer [56]	Transformer Encoder, CNNs Stem, Multi-Dilated CNNs Decoder	3	114M
RetinaFace [17]	ResNet,FPN,ClassficationHead,RegressionHead,ContextModule	5	30M
Generalist Models			
UniTab [58]	Image Encoder,Text Encoder, Multimodal Encoder, Decoder	4	185M
Uni-Perceiver [64]	None	1	124M
Uni-Perceiver-MoE [62]	None	1	167M
Uni-Perceiver-V2 [32]	ResNet,RPN,Mask DINO,RoBERTa,Decoder,ClassficationHead,RegressionHead,MaskHead	8	308M
Pix2Seq v2 [11]	ViT,Decoder	2	132M
Unified-IO$_{XL}$ [39]	VQ-VAE Encoder,VQ-VAE Decoder,Encoder,Decoder	4	2.9B
Shikra-13B [8]	ViT,Vicuna,Image Adapter	3	13B
Ferret-13B [59]	ViT,Vicuna,Visual Sampler,KNN	4	13B
VisionLLM-R50 [53]	ResNet,Language-Guided Image Tokenizer,Encoder,Decoder,Alpaca-7B	5	7B
GLIP-T [35]	Swin,FPN,Text Encoder,Dy-Head,Fusion Module	5	431M
Grounding DINO-T [37]	Swin,DINO,BERT,Feature Enhancer,Decoder,Query Selection	6	174M
BLIP (129M) [34]	ViT-L,BERT,Image-grounded Text Encoder, Image-grounded Text Decoder	4	583M
BLIP-2 (129M) [33]	ViT-G,Qformer,Adapter,LLM	4	12.1B
ReCo+ [49]	DeiT-SIN,CLIP,DenseCLIP,DeepLabV3+	4	46M
XDecoder(T) [65]	FocalNet,Encoder,Decoder,Latent Query	4	165M

various other tasks by incorporating relevant data.

Negative Societal Impact. Our largest model necessitates 7 days of training on 96 A100 GPUs, leading to considerable carbon emissions. Furthermore, the generated content might reflect biases from the training data, stemming from a lack of alignment with human preferences.

References

1. 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)
2. 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`
3. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: CVPR (2020)
4. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: CVPR (2018)
5. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)

Fig. 3: Visualizations on cross-attention between task token and image, with yellower colors indicating higher responses.

6. 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)
7. Chen, C., Borgeaud, S., Irving, G., Lespiau, J.B., Sifre, L., Jumper, J.: Accelerating large language model decoding with speculative sampling. arXiv preprint arXiv:2302.01318 (2023)
8. 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)
9. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: ECCV (2018)
10. Chen, T., Saxena, S., Li, L., Fleet, D.J., Hinton, G.: Pix2seq: A language modeling framework for object detection. In: ICLR (2022)
11. Chen, T., Saxena, S., Li, L., Lin, T.Y., Fleet, D.J., Hinton, G.E.: A unified sequence interface for vision tasks. NeurIPS (2022)
12. 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)
13. 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)
14. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: CVPR (2022)
15. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. In: ICML (2021)
16. 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)
17. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S.: Retinaface: Single-shot multi-level face localisation in the wild. In: CVPR (2020)
18. 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)
19. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV (2010)
20. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML. PMLR (2017)
21. Gan, Z., Chen, Y.C., Li, L., Zhu, C., Cheng, Y., Liu, J.: Large-scale adversarial training for vision-and-language representation learning. NeurIPS (2020)
22. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: CVPR (2019)
23. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
24. III/4, I.W.: ISPRS 2D Semantic Labeling Contest, https://www.isprs.org/education/benchmarks/UrbanSemLab/2d-sem-label-potsdam.aspx
25. 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)
26. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR (2015)
27. Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.: Referitgame: Referring to objects in photographs of natural scenes. In: EMNLP (2014)
28. 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)
29. 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)
30. Kuhn, H.W.: The hungarian method for the assignment problem. NRL (1955)
31. 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)
32. 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)
33. 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)
34. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML (2022)
35. 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)
36. 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)
37. 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)
38. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In: CVPR (2016)
39. 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)
40. Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A.L., Murphy, K.: Generation and comprehension of unambiguous object descriptions. In: CVPR (2016)

41. Mottaghi, R., Chen, X., Liu, X., Cho, N.G., Lee, S.W., Fidler, S., Urtasun, R., Yuille, A.: The role of context for object detection and semantic segmentation in the wild. In: CVPR (2014)
42. Neuhold, G., Ollmann, T., Rota Bulo, S., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: ICCV (2017)
43. Ordonez, V., Kulkarni, G., Berg, T.: Im2text: Describing images using 1 million captioned photographs. NeurIPS **24** (2011)
44. 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)
45. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. NeurIPS (2015)
46. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015)
47. 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)
48. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: ACL (2018)
49. Shin, G., Xie, W., Albanie, S.: Reco: Retrieve and co-segment for zero-shot transfer. In: NeurIPS (2022)
50. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: CVPR (2015)
51. 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)
52. 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)
53. 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)
54. Wang, Y., Chen, X., Cao, L., Huang, W., Sun, F., Wang, Y.: Multimodal token fusion for vision transformers. In: CVPR (2022)
55. 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)
56. 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)
57. Yang, S., Luo, P., Loy, C.C., Tang, X.: Wider face: A face detection benchmark. In: CVPR (2016)
58. 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)
59. You, H., Zhang, H., Gan, Z., Du, X., Zhang, B., Wang, Z., Cao, L., Chang, S.F., Yang, Y.: Ferret: Refer and ground anything anywhere at any granularity. In: ICLR (2024)
60. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In: CVPR (2020)
61. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: CVPR (2017)

62. 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)
63. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. ICLR (2020)
64. 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)
65. 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)

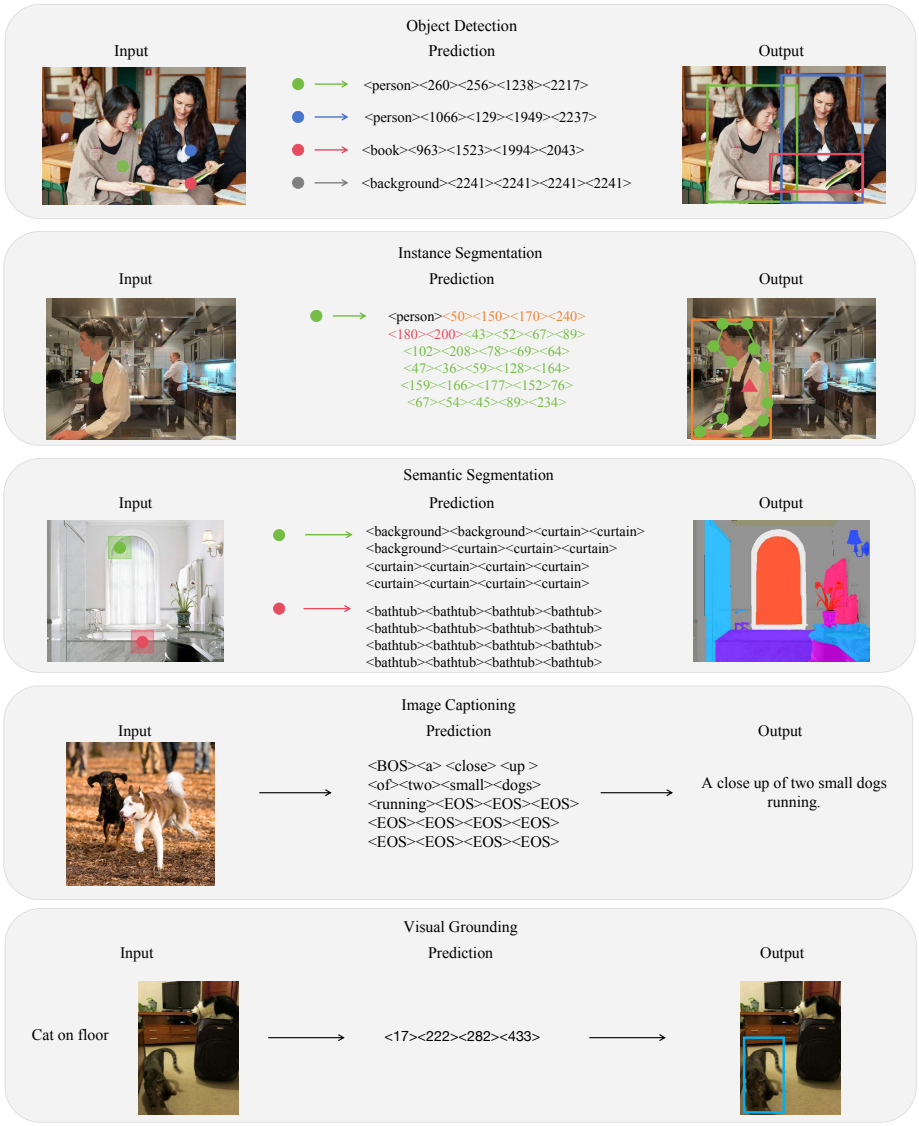

Fig. 4: Visualization of five standard vision-centric tasks.

Fig. 5: Qualitative results on zero-shot datasets. Zoom in for better viewing.

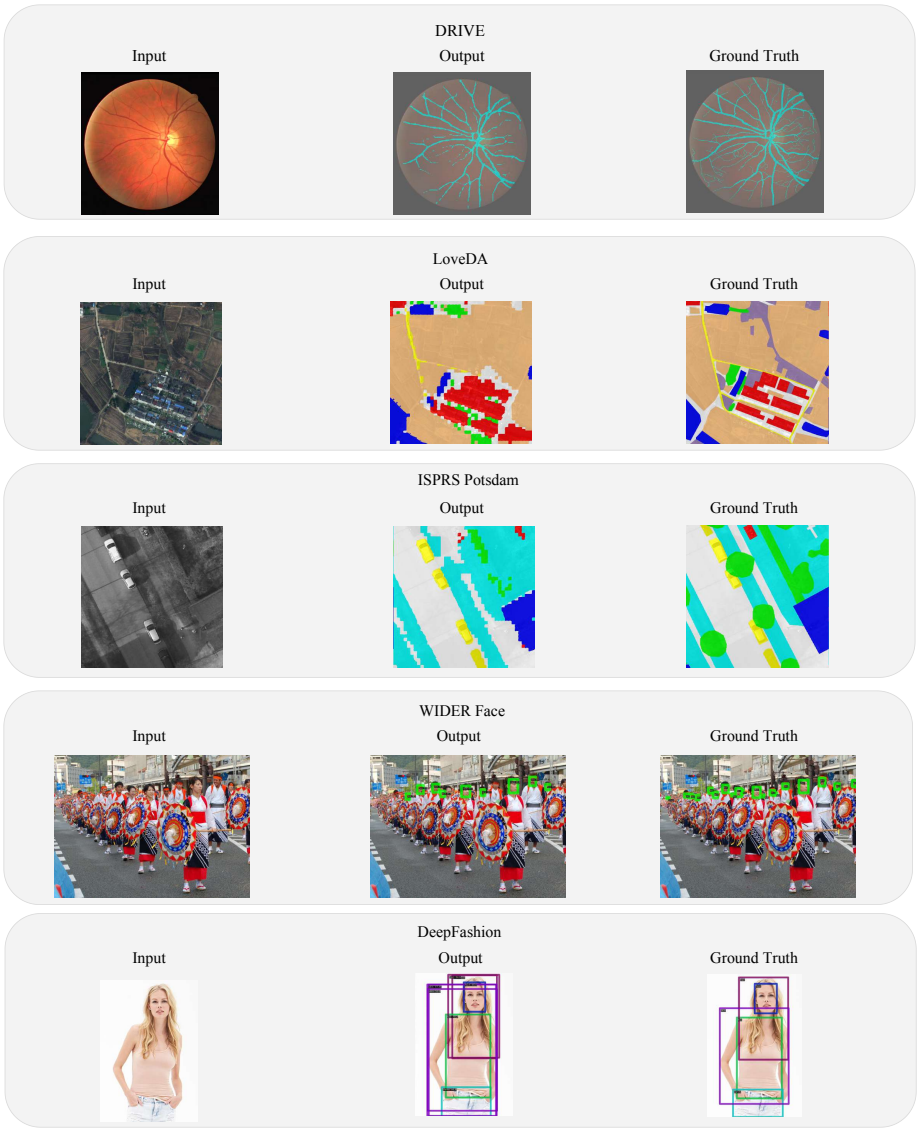

Fig. 6: Qualitative results on few-shot datasets. Zoom in for better viewing.