Supplementary Materials

A. Comprehensive Comparison of Image Classification Performance

Due to space limitations in the main body of the paper, we present a more comprehensive comparison of image classification performance in Table A.

B. Detailed Training Settings

In this section, we present the detailed training recipes for image classification, object detection, and semantic segmentation.

B.1. Settings for Backbone-Level Comparison

ImageNet image classification. The training details of image classification on ImageNet [18] are shown in Table B, which are similar to common practices [1, 3, 7, 10] and with some tweaks. To further explore the capability of our model and match the large-scale private data used in previous methods [4, 8, 16], we adopt M3I Pre-training [20], a unified pre-training approach available for both unlabeled and weakly-labeled data, to pre-train InternImage-H on a 427 million joint dataset of public Laion-400M [21], YFCC-15M [22], and CC12M [23] for 30 epochs, and then we fine-tune the model on ImageNet-1K for 20 epochs. For the more detailed pre-training settings of InternImage-H, please refer to M3I Pre-training [20].

COCO object detection. We verify the detection performance of our InternImage on the COCO benchmark [24], on top of Mask R-CNN [25] and Cascade Mask R-CNN [26]. For fair comparisons, we follow common practices [3, 6] to initialize the backbone with pre-trained classification weights, and train these models using a $1\times$ (12 epochs) or $3\times$ (36 epochs) schedule by default. For $1\times$ schedule, the image is resized to have a shorter side of 800 pixels, while the longer side does not exceed 1,333 pixels. During testing, the shorter side of the input image is fixed to 800 pixels. For $3\times$ schedule, the shorter side is resized to $480-800$ pixels, while the longer side does not exceed 1,333 pixels. All these detection models are trained with a batch size of 16 and optimized by AdamW [27] with an initial learning rate of 1×10^{-4}.

ADE20K semantic segmentation. We evaluate our InternImage models on the ADE20K dataset [28], and initialize them with the pre-trained classification weights. For the InternImage-T/S/B models, we optimize them using AdamW [27] with an initial learning rate of 6×10^{-5}, and 2×10^{-5} for InternImage-X/XL. The learning rate is decayed following the polynomial decay schedule with a power of 1.0. Following previous methods [3, 6, 10], the crop size is set to 512 for InternImage-T/S/B, and 640 for

method	type	scale	#params	#FLOPs	acc (%)
DeiT-S [1]	T	224^2	22M	5G	79.9
PVT-S [2]	T	224^2	25M	4G	79.8
Swin-T [3]	T	224^2	29M	5G	81.3
CoAtNet-0 [4]	T	224^2	25M	4G	81.6
CSwin-T [5]	T	224^2	23M	4G	82.7
PVTv2-B2 [6]	T	224^2	25M	4G	82.0
DeiT III-S [7]	T	224^2	22M	5G	81.4
SwinV2-T/8 [8]	T	256^2	28M	6G	81.8
Focal-T [9]	T	224^2	29M	5G	82.2
ConvNeXt-T [10]	C	224^2	29M	5G	82.1
ConvNeXt-T-dcls [11]	C	224^2	29M	5G	82.5
SLaK-T [12]	C	224^2	30M	5G	82.5
HorNet-T [13]	C	224^2	23M	4G	83.0
InternImage-T (ours)	C	224^2	30M	5G	83.5
PVT-L [2]	T	224^2	61M	10G	81.7
Swin-S [3]	T	224^2	50M	9G	83.0
CoAtNet-1 [4]	T	224^2	42M	8G	83.3
PVTv2-B4 [6]	T	224^2	63M	10G	83.6
SwinV2-S/8 [8]	T	256^2	50M	12G	83.7
ConvNeXt-S [10]	C	224^2	50M	9G	83.1
SLaK-S [12]	C	224^2	55M	10G	83.8
HorNet-S [13]	C	224^2	50M	9G	84.0
InternImage-S (ours)	C	224^2	50M	8G	84.2
DeiT-B [1]	T	224^2	87M	18G	83.1
Swin-B [3]	T	224^2	88M	15G	83.5
CoAtNet-2 [4]	T	224^2	75M	16G	84.1
PVTv2-B5 [6]	T	224^2	82M	12G	83.8
DeiT III-B [7]	T	224^2	87M	18G	83.8
SwinV2-B/8 [8]	T	256^2	88M	20G	84.2
RepLKNet-31B [14]	C	224^2	79M	15G	83.5
ConvNeXt-B [10]	C	224^2	88M	15G	83.8
SLaK-B [12]	C	224^2	95M	17G	84.0
HorNet-B [13]	C	224^2	88M	16G	84.3
InternImage-B (ours)	C	224^2	97M	16G	84.9
Swin-L‡ [3]	T	384^2	197M	104G	87.3
CoAtNet-4‡ [4]	T	384^2	275M	190G	87.9
DeiT III-L‡ [7]	T	384^2	304M	191G	87.7
SwinV2-L/24‡ [8]	T	384^2	197M	115G	87.6
RepLKNet-31L‡ [14]	C	384^2	172M	96G	86.6
HorNet-L‡ [13]	C	384^2	202M	102G	87.7
ConvNeXt-L‡ [10]	C	384^2	198M	101G	87.5
ConvNeXt-XL‡ [10]	C	384^2	350M	179G	87.8
InternImage-L‡ (ours)	C	384^2	223M	108G	87.7
InternImage-XL‡ (ours)	C	384^2	335M	163G	88.0
ViT-G/14$^\#$ [15]	T	518^2	1.84B	5160G	90.5
CoAtNet-6$^\#$ [4]	T	512^2	1.47B	1521G	90.5
CoAtNet-7$^\#$ [4]	T	512^2	2.44B	2586G	90.9
Florence-CoSwin-H$^\#$ [16]	T	–	893M	–	90.0
SwinV2-G$^\#$ [8]	T	640^2	3.00B	–	90.2
RepLKNet-XL$^\#$ [14]	C	384^2	335M	129G	87.8
BiT-L-ResNet152x4$^\#$ [17]	C	480^2	928M	–	87.5
InternImage-H$^\#$ (ours)	C	224^2	1.08B	188G	88.9
InternImage-H$^\#$ (ours)	C	640^2	1.08B	1478G	89.6

Table A. **Image classification performance on the ImageNet validation set**. "type" refers to model type, where "T" and "C" denote transformer and CNN, respectively. "scale" is the input scale. "acc" is the top-1 accuracy. "$\ddagger$" indicates the model is pre-trained on ImageNet-22K [18]. "#" indicates pretraining on extra large-scale private dataset such as JFT-300M [19], FLD-900M [16], or the joint public dataset in this work.

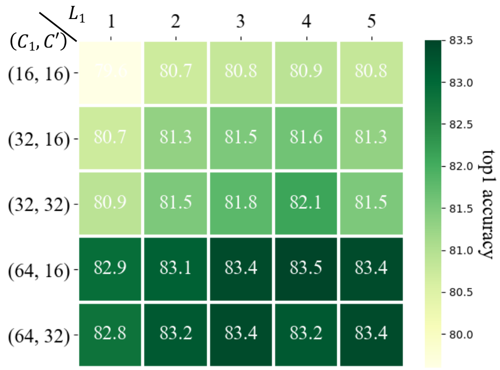

Figure A. **Comparison of different stacking hyper-parameters**. Each square indicates the accuracy of the model determined by hyperparameter, with the darker the color, the higher the accuracy.

InternImage-L/XL. All segmentation models are trained using UperNet [29] with a batch size of 16 for 160k iterations, and compared fairly with previous CNN-based and transformer-based backbones.

B.2. Settings for System-Level Comparison

COCO object detection. For system-level comparison with state-of-the-art large-scale detection models [8, 30–33], we first initialize the InternImage-XL/H backbone with the weights pre-trained on ImageNet-22K or the 427M large-scale joint dataset, and double its parameters using the composite techniques [33]. Then, we pre-train the model along with the DINO [31] detector on the Objects365 [34] for 26 epochs, with an initial learning rate of 2×10^{-4} and a batch size of 256. The shorter size of input images is resized to $600-1200$ pixels during pre-training, and the learning rate drops by 10 times at epoch 22. Finally, we fine-tune these detectors on the COCO dataset for 12 epochs, where the batch size is 64, and the initial learning rate is 5×10^{-5}, which drops by 10 times at the final epoch.

ADE20K semantic segmentation. To further reach leading segmentation performance, we first initialize our InternImage-H backbone with the pre-trained weights on the 427M large-scale joint dataset, and arm it with the state-of-the-art segmentation method Mask2Former [35]. We follow the same training settings in [30, 36], *i.e.* pre-training and fine-tuning the model on COCO-Stuff [37] and ADE20K [28] datasets both for 80k iterations, with a crop size of 896 and an initial learning rate of 1×10^{-5}.

C. Exploration of Hyper-parameters

C.1. Model Stacking

As discussed in Section 3.2, our model is constructed in four stacking rules, and we further restrict the model parameters to 30M for the origin model. We discretize the stacking hyper-parameters C_1 to $\{16, 32, 64\}$, L_1 to $\{1, 2, 3, 4, 5\}$, and C' to $\{16, 32\}$. And L_2 is determined by selecting the model size to approximately 30M. In this way, we obtained 30 models by combining the three hyper-parameters.

We adopt the training recipe listed in Table B to train our -T models unless otherwise stated. Fig. A shows the ImageNet-1K top-1 accuracy of these models under the same training settings, with darker green indicating higher accuracy, *i.e.*, models with stronger representational capability. When C' equals 16, models are generally higher than that with C' of 32, and L_1 works best at 4, thanks to a reasonable stacking ratio. A large number of channels allows for more gain. Finally, through the above exploration experiments, we determine our basic stacking hyper-parameter (C_1, C', L_1, L_3) to $(64, 16, 4, 18)$.

C.2. Model Scaling

In Section 3.2, we have shown the constraints on the depth scaling factor α and the width scaling factor β. Based on this condition and the -T model (30M), we display reasonable scaling possibilities for extending the -T model to -B models (100M). As illustrated in Table C, the first two columns show the formulas for α and β. The penultimate column indicates model parameters, and the last column indicates the ImageNet-1K top-1 accuracy of these models after 300 training epochs.

It is worth noting that the model width C_1 needs to be divisible by C'. Therefore some adjustment is required in determining the specific scaling parameters. This results in a small fluctuation in the number of parameters, but this is acceptable. Our exploratory experiments prove that when (α, β) is set at $(1.09, 1.36)$ for the best performance. In addition, the other size models -S/L/XL/H also confirmed the effectiveness of our scaling rules.

C.3. Kernel Size

As mentioned in Section 3.1, we argue 3×3 dynamic sparse convolution is enough for the large receptive field. Here, we explore the role played by the number of convolutional neurons in the DCNv3 operator. Specifically, we replaced the 3×3 kernel in the DCNv3 operator with the 5×5 or 7×7 kernel. They are all trained by the -T training recipes (see Table B) and validated on the ImageNet-1K validation set. The results are shown in Table D.

The results show that when enlarging the convolution kernel, the parameters and FLOPs are followed by the surge, while the accuracy is not significantly improved (83.5 *v.s* 83.6) or even decreased (83.5 *v.s* 82.8). These results show that when the number of convolutional neurons in a single layer increases, the model becomes more difficult to optimize. This phenomenon is also confirmed in RepLKNet [14], and it addresses this problem by re-

settings	InternImage-T	InternImage-S	InternImage-B	InternImage-L		InternImage-XL		InternImage-H
	IN-1K pt	IN-1K pt	IN-1K pt	IN-22K pt	IN-1K ft	IN-22K pt	IN-1K ft	IN-1K ft
input scale	224	224	224	192	384	192	384	224/640
batch size	4096	4096	4096	4096	512	4096	512	512
optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW
LR	4×10^{-3}	4×10^{-3}	4×10^{-3}	1×10^{-3}	2×10^{-5}	1×10^{-3}	2×10^{-5}	2×10^{-5}
LR schedule	cosine	cosine	cosine	cosine	cosine	cosine	cosine	cosine
weight decay	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
warmup epochs	5	5	5	5	0	5	0	0
epochs	300	300	300	90	20	90	20	20
horizontal flip	✓	✓	✓	✓	✓	✓	✓	✓
random resized crop	✓	✓	✓	✓	✓	✓	✓	✓
auto augment	✓	✓	✓	✓	✓	✓	✓	✓
layer scale	✗	✓	✓	✓	✓	✓	✓	✓
mixup alpha	0.8	0.8	0.8	0.8	✗	0.8	✗	✗
cutmix alpha	1.0	1.0	1.0	1.0	✗	1.0	✗	✗
erasing prob.	0.25	0.25	0.25	0.25	✗	0.25	✗	✗
color jitter	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
label smoothing ε	0.1	0.1	0.1	0.1	0.3	0.1	0.3	0.3
dropout	✗	✗	✗	✗	✗	✗	✗	✗
drop path rate	0.1	0.4	0.5	0.1	0.1	0.2	0.2	0.2
repeated aug	✗	✗	✗	✗	✗	✗	✗	✗
gradient clip	5.0	5.0	5.0	5.0	5.0	5.0	5.0	5.0
loss	CE	CE	CE	CE	CE	CE	CE	CE

Table B. **Detailed training recipe for InternImage of different parameter scales on ImageNet [18].** "CE" denotes the cross entropy loss, "LR" denotes the learning rate. The training recipe follows common practices [1, 3, 7, 10] and has some tune-ups. "IN-1K pt", "IN-22K pt", and "IN-1K ft" represent ImageNet-1K pre-training, ImageNet-22K pre-training, and ImageNet-1K fine-tuning, respectively.

scaling factors α	scaling factors β	#parameters	top-1 accuracy (%)
1.03	1.40	118M	84.5
1.06	1.38	95M	83.8
1.09	1.36	97M	84.9
1.12	1.34	105M	83.1
1.15	1.32	95M	81.8

Table C. **Comparison of different scaling factors**. The default setting is marked with a gray background.

kernel size	#parameters	FLOPs	top-1 accuracy (%)
3×3	30M	5G	83.5
5×5	37M	6G	83.6
7×7	48M	8G	82.8

Table D. **Comparison of different kernel sizes in our operator**. The default setting is marked with a gray background.

parameterizing [14] techniques, which might bring extra time and memory costs in the training phase. In this work, we avoid this problem by adopting the simple yet effective 3×3 DCNv3 as InternImage's core operator.

Fig. B shows the effective receptive fields (ERF) of ResNet-101 [38] and InternImage-S. A wider distribution of bright areas indicates a larger ERF. We uniformly activate the input image at the dog's eye, count the gradient map of each block, aggregate by channel, and map back to the input image. We see that the ERF of ResNet-101 [38] without training is limited to a local area, while the fully trained ResNet-101 still has an ERF around the eye, and the gradient amplitude is lower, and the distribution is more sparse. Therefore, the area that ResNet-101 can effectively perceive is very limited. For the InternImage-S without training, its ERF is concentrated around the activation point. Since the offset is not learned, its ERF is also very small in the last two blocks. But after sufficient training, InternImage-L can effectively perceive the information of the entire image in the 3-rd and 4-th stages.

D. Additional Downstream Tasks

D.1. Classification

iNaturalist 2018 [51] is a read-word long-tailed dataset containing 8142 fine-graned species. The dataset comprises 437.5K training images and an imbalance factor of 500. For this experiment, we initialize our InternImage-H model with the pre-trained weights on the 427M large-scale joint dataset, and fine-tune it on the training set of iNaturalist 2018 for 100 epochs. We follow MetaFormer [39] to adopt a resolution of 384×384 for fine-tuning, with the utilization of meta information. Other training settings are the same as the recipe for fine-tuning InternImage-H on ImageNet-1K, as reported in Table B. As a result, our method achieves the state-of-the-art accuracy of 92.6 (see Table E) on the validation set of iNaturalist 2018, 3.9 points better than the previous best model MetaFormer [39].

Places205 [52] is a dataset containing 2.5 million images of 205 scene categories, which are dedicated to the scene recognition task. The images in this dataset cover a

method	classification			semantic segmentation				
	iNaturalist2018	Places205	Places365	COCO-Stuff-10K	Pascal Context	Cityscapes (val)	Cityscapes (test)	NYU Depth V2
previous best	88.7[a]	69.3[b]	60.7[c]	54.2[d]	68.2[d]	86.9[e]	85.2[d]	56.9[f]
InternImage-H	92.6 (+3.9)	71.7 (+2.4)	61.2 (+0.5)	59.6 (+5.4)	70.3 (+2.1)	87.0 (+0.1)	86.1 (+0.9)	68.1 (+11.2)

method	object detection						
	LVIS (minival)	LVIS (val)	VOC2007	VOC2012	OpenImages	CrowdHuman	BDD100K
previous best	59.8[g]	62.2[h]	89.3[i]	92.9[j]	72.2[k]	94.1[l]	35.6[m]
InternImage-H	65.8 (+6.0)	63.2 (+1.0)	94.0 (+4.7)	97.2 (+4.3)	74.1 (+1.9)	97.2 (+3.1)	38.8 (+3.2)

Table E. **Summary of InternImage-H performance on various mainstream vision benchmarks**. a: MetaFormer [39]. b: MixMIM-L [40]. c: SWAG [41]. d: ViT-Adapter [36]. e: PSA [42]. f: CMX-B5 [43]. g: GLIPv2 [44]. h: EVA [45]. i: Cascade Eff-B7 NAS-FPN [46]. j: ATLDETv2 [47]. k: OpenImages 2019 competition 1st [48]. l: Iter-Deformable-DETR [49]. m: PP-YOLOE [50].

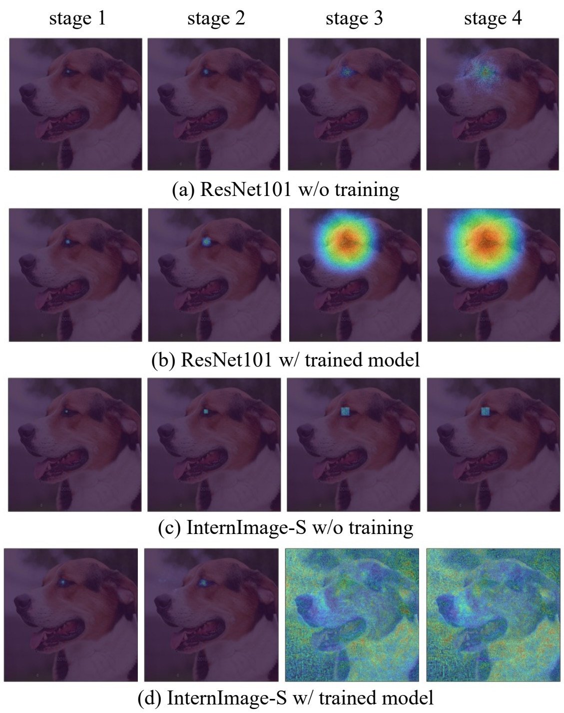

Figure B. **Visualization of the effective receptive field (ERF) of different backbones.** The activated pixel is at dog's eye. (a) and (b) shows the ERF of ResNet-101 [38] with (w/) and without (w/o) training on ImageNet-1K [18], respectively. (c) and (d) are the ERF of InternImage-B with (w/) and without (w/o) training on ImageNet-1K.

wide range of indoor and outdoor scenes, such as offices, kitchens, forests, and beaches. We initialize our model with pre-trained weights on a large-scale joint dataset, consisting of 427 million images, and fine-tune it on the Places205 training set. Other training settings are the same as the recipe for fine-tuning InternImage-H on ImageNet-1K, as reported in Table B. Our method achieves state-of-the-art accuracy of 71.7 (see Table E) on the validation set of Places205, outperforming the previous best model MixMIM-L [40] by 2.4 points.

Places365 [53] is a dataset containing 1.8 million images of 365 scene categories, which are dedicated to the scene recognition task. The images in this dataset cover a wide range of indoor and outdoor scenes, such as airports, bedrooms, deserts, and waterfalls. The specific pre-training and fine-tuning strategies are the same as for Places205. Our method achieves state-of-the-art accuracy of 61.2 (see Table E) on the validation set of Places365, outperforming the previous best model SWAG [41] by 0.5 points. The Places365 dataset provides a more fine-grained classification task compared to Places205, allowing our model to learn more subtle differences between similar scenes.

D.2. Object Detection

LVIS v1.0 [54] is a large-scale vocabulary dataset for object detection and instance segmentation tasks, which contains 1203 categories in 164k images. For this dataset, we initialize our InternImage-H with the Objects365 [34] pre-trained weights, then fine-tune it on the training set of LVIS v1.0. Here, we report the box AP (*i.e.*, AP^{b}) with multi-scale testing on the minival set and the val set, respectively. As shown in Table E, our InternImage-H creates a new record of 65.8 AP^{b} on the LVIS minival, and 63.2 AP^{b} on the LVIS val, outperforming previous state-of-the-art methods by clear margins.

Pascal VOC [55] contains 20 object classes, which has been widely used as a benchmark for object detection tasks. We adopt this dataset to further evaluate the detection performance of our model. Specifically, we employ the Objects365 [34] pre-trained weights to initialize our InternImage-H, and fine-tune it on the trainval set of Pascal VOC 2007 and Pascal VOC 2012 following previous method [46]. As shown in Table E, on the Pascal VOC 2007 test set, our InternImage-H yields 94.0 AP^{50} with single-scale testing, which is 4.7 points better than previous best Cascade Eff-B7 NAS-FPN [46]. On the Pascal VOC 2012 test set, our method achieves 97.2 mAP, 4.3 points higher than the best record on the official leaderboard [47].

OpenImages v6 [56] is a dataset of about 9 million images with 16M bounding boxes for 600 object classes on 1.9 million images dedicated to the object detection task, which are very diverse and often embrace complex scenes with multiple objects (8.3 per image on average). For this dataset, we use the same settings as the previous two datasets. In addition, we follow [48] to use the class-aware sampling during fine-tuning. As reported in Table E, our InternImage-H yields 74.1 mAP, achieving 1.9 mAP improvement compared to the previous best results [48].

CrownHuman [57] is a benchmark dataset to better evaluate detectors in crowd scenarios. The CrowdHuman dataset is large, rich-annotated and contains high diversity. CrowdHuman contains 15000, 4370 and 5000 images for training, validation, and testing, respectively. There are a total of 470K human instances from train and validation subsets and 23 persons per image, with various kinds of occlusions in the dataset. We used the same training setup as for the previous dataset. Our pre-trained model reached optimal performance in 3750 iterations, exceeding the previous best model Iter-Deformable-DETR [49] by 3.1 AP.

BDD100K [58] is a dataset of around 100K high-resolution images with diverse weather and lighting conditions, containing 10 object categories, including pedestrians, cars, buses, and bicycles, dedicated to the object detection task. The images in this dataset are captured from a moving vehicle, simulating real-world scenarios. For this experiment, we initialize our InternImage-H model with the pre-trained weights on the 427M joint dataset and fine-tune it on the BDD100K training set for 12 epochs. As reported in Table E, our InternImage-H achieves 38.8 mAP on the validation set, which is the state-of-the-art performance, surpassing the previous best model by 3.2 mAP. Our method demonstrates superior performance in detecting objects in real-world driving scenarios, which can benefit autonomous driving and intelligent transportation systems.

D.3. Semantic Segmentation

COCO-Stuff [37] includes the images from the COCO [24] dataset for semantic segmentation, spanning over 171 categories. Specifically, COCO-Stuff-164K is the full set that contains all 164k images, while COCO-Stuff-10K is a subset of the -164K that splits into 9,000 and 1,000 images for training and testing. Here, we equip our InternImage-H with the advanced Mask2Former [35], and pre-train the model on the COCO-Stuff-164K for 80k iterations. Then we fine-tune it on the COCO-Stuff-10K for 40k iterations and report the multi-scale mIoU. The crop size is set to 512×512 in this experiment. As shown in Table E, our model achieves 59.6 MS mIoU on the test set, outperforming the previous best ViT-Adapter [36] by 5.4 mIoU.

Pascal Context [59] contains 59 semantic classes. It is divided into 4,996 images for training and 5,104 images for testing. For this dataset, we also employ Mask2Former with our InternImage-H, and follow the training settings in [36]. Specifically, we first load the classification pre-trained weights to initialize the model, then fine-tune it on the training set of Pascal Context for 40k iterations. The crop size is set to 480×480 in this experiment. As shown in Table E, our method reports 70.3 MS mIoU on the test set, which is 2.1 points better than ViT-Adapter [36].

Cityscapes [60] is a high-resolution dataset recorded in street scenes including 19 classes. In this experiment, we use Mask2Former [35] as the segmentation framework. Following common practices [36, 61, 62], we first pre-train on Mapillary Vistas [63] and then fine-tune on Cityscapes for 80k iterations, respectively. The crop size is set to 1024×1024 in this experiment. As shown in Table E, our InternImage-H achieves 87.0 MS mIoU on the validation set, and 86.1 MS mIoU on the test set.

NYU Depth V2 [64] comprises of 1449 RGB-D images, each with a size of 640×480. These images are divided into 795 training and 654 testing images, each with annotations on 40 semantic categories. We adopt the same training settings as we used when fine-tuning on Pascal Context. As shown in Table E, our method achieves a big jump to 68.1 MS mIoU on the validation set, which is 11.2 points better than CMX-B5 [43].

method	#params	scale	FLOPs	acc (%)	throughput (img/s)
InternImage-B (ours)	97M	224^2	16G	84.9	775
		800^2	206G	–	54
InternImage-B-DCNv2 [65]	146M	224^2	24G	–	311
		800^2	313G	–	16
ConvNeXt-B [10]	88M	224^2	15G	83.8	881
		800^2	196G	–	58
RepLKNet-B [14]	79M	224^2	15G	83.5	884
		800^2	198G	–	21
DAT-B [10]	88M	224^2	16G	84.0	661
		800^2	194G	–	24

Table F. **Throughput comparison of different models under different input resolutions.** "#params" denotes the number of parameters. "acc" represents the top-1 accuracy on the ImageNet-1K validation set. The throughputs of 224×224 and 800×800 input resolutions are tested with the batch size of 256 and 2 respectively, using a single A100 GPU.

method	#params	scale	GFLOPs	throughput	memory	acc (%)	mAP
InternImage-L	223M	$384^2/800^2$	108/469	148/33	39G/43G	87.7/-	-/56.0
Swin-L [3]	197M	$384^2/800^2$	104/451	183/39	35G/38G	87.3/-	-/53.9

Table G. **Efficiency comparison with Swin-Transformer.** Throughput (img/s) is measured on an A100 GPU. Throughtput and memory are measured with a batch size of 16 for 384^2 and 4 for 800^2. The mAP refers to the bounding box mAP with Cascade R-CNN ($3\times$ + MS) on COCO.

E. Throughput Analysis

In this section, we benchmark the throughput of our InternImage with counterparts, including a variant equipped with DCNv2 [65], ConvNext [10], RepLKNet [14], and a vision transformer with deformable attention (DAT) [66].

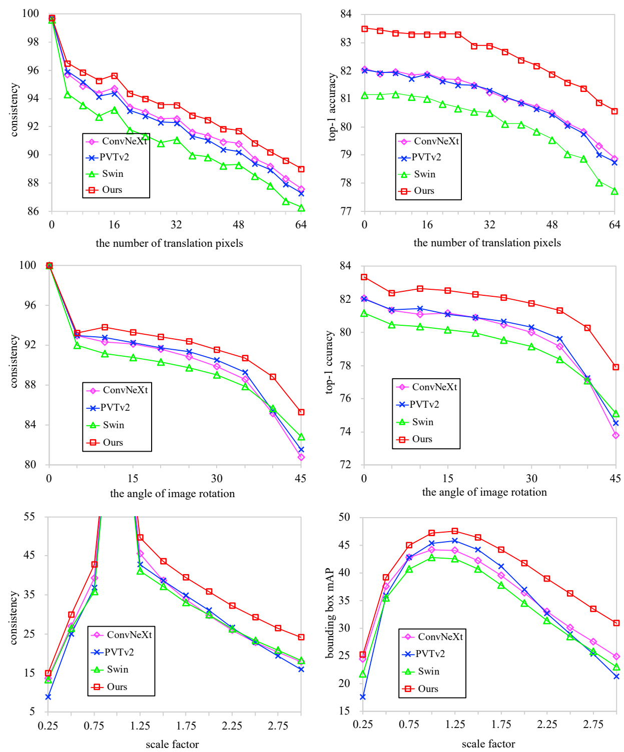

Figure C. **Comparison of robust evaluation of different methods.** These results show that our model has better robustness in terms of translation, rotation, and input resolution.

As shown in Table F, compared to the variant with DCNv2 [65], our model enjoys better parameter-efficient and significantly faster inference speed under both 224×224 and 800×800 input resolutions. Compared to RepLKNet-B [14] and DAT-B [66], our model has a throughput advantage at a high input resolution (*i.e.*, 800×800). This resolution is widely used in dense prediction tasks such as object detection. Compared with ConvNeXt [10], despite the throughput gap due to DCN-based operators, our model still has an accuracy advantage (84.9 *vs*. 83.8), and we are also looking for an efficient DCN to make our model more suitable for downstream tasks that require high efficiency. In Table G, we provide a full comparison of InternImage with Swin Transformer [3] in terms of throughput and memory, which shows that InternImage obtains better accuracy than Swin-L on various tasks with comparable inference efficiency.

F. Robustness Evaluation on ImageNet

In this section, we evaluate the robustness of different models under different transformations (see Fig. C). We consider translation, rotation, and scaling to evaluate. The models we choose for comparison include a convolutional model (ConvNeXt-T [10]), a local attention-based model (Swin-T [3]), a global attention-based model (PVTv2-B2 [6]), and our InternImage-T.

F.1. Translation Invariance

Translation invariance describes the capability of the model to retain the original output when the input image is translated. We evaluate the translation invariance in the classification task by dithering the image from 0 to 64 pixels. The invariance is measured by the probability that the model predicts the same label when the same input image is translated. The first row of Fig. C indicates our InternImagehas the translation invariance of the different methods. It is evident that the robustness of the four models to translation is shown as our method is the best, followed by convolution-based ConvNeXt, followed by global attention-based PVTv2, and the worst local attention-based Swin Transformer [3].

F.2. Rotation Invariance

To evaluate the rotation invariance of the classification task, we rotate the image from 0° to 45° in steps of 5°. In a similar way to translation invariance, the predicted consistency under different rotation angles is used to evaluate the rotational invariance. From the second row of Fig. C, we found that the consistency performance of all models is comparable in the small angle phase. However, at large-angle rotation (*i.e.*, $> 10°$), our model is clearly superior to the other models.

F.3. Scaling Invariance

We evaluate the scaling invariance on object detection. The scaling factor of the input image varies from 0.25 to 3.0 in steps of 0.25. Detection consistency is defined as the invariance metric for the detection task. The predicted boxes on the scaled images are first converted back to the original resolution, and then the predicted boxes at the original resolution are used as the ground truth boxes to calculate the box mAP. As seen in the last row of Fig. C, we can observe that all methods of our experiments are sensitive to downscaling. And they show invariance comparable to the input at small resolutions. Our method performs better when scaling up the images. Both box consistency and bounding box mAP are better than the others.

F.4. How Hungry the Model is for Data Scale?

In order to verify the robustness of the model to the data scale. We uniformly sampled the ImageNet-1K data to obtain 1%, 10%, and 100% data, respectively. And we chose ResNet-50 [38], ConvNeXt-T [10], Swin-T [3], InternImage-T-dattn and our InternImage-T to conduct 300 rounds of training experiments on these data. The experimental settings are consistent with Table B. The experimental results can be viewed in Table H. We see that ResNet [38] performs best on the 1% and 10% data (12.2%

method	1%	10%	100%
ResNet-50 [38]	12.2	57.5	80.4
ConvNeXt-T [10]	8.4	52.6	82.1
Swin-T [3]	failed	12.1	81.3
InternImage-T-dattn [67]	4.1	49.9	81.9
InternImage-T (ours)	5.9	56.0	83.5

Table H. **Accuracy of different models at different data scales**. "InternImage-dattn" refers to the model variant equipped with deformable attention [67].

& 57.5%), benefiting from its inductive biases. But its upper limitation is low (80.4%) when the data is sufficient. Swin-T fails completely in 1% datasets and shows good performance only on the 100% dataset. The proposed InternImage-T has strong robustness not only on 1% and 10% data (5.9% and 56.0%) but also on full data (83.5%), which is consistently better than the InternImage-T variant with deformable attention (dattn) and ConvNeXt [10]. These results indicate the robustness of our model with respect to the data scale.

References

[1] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning.*, pages 10347–10357, 2021. 1, 3

[2] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Int. Conf. Comput. Vis.*, pages 568–578, 2021. 1

[3] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, pages 10012–10022, 2021. 1, 3, 5, 6, 7

[4] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *Adv. Neural Inform. Process. Syst.*, 34:3965–3977, 2021. 1

[5] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12124–12134, 2022. 1

[6] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 1, 6

[7] Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. *arXiv preprint arXiv:2204.07118*, 2022. 1, 3

[8] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. *Adv. Neural Inform. Process. Syst.*, pages 12009–12019, 2022. 1, 2

[9] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *arXiv preprint arXiv:2107.00641*, 2021. 1

[10] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *arXiv preprint arXiv:2201.03545*, 2022. 1, 3, 5, 6, 7

[11] Ismail Khalfaoui Hassani, Thomas Pellegrini, and Timothée Masquelier. Dilated convolution with learnable spacings. *arXiv preprint arXiv:2112.03740*, 2021. 1

[12] Shiwei Liu, Tianlong Chen, Xiaohan Chen, Xuxi Chen, Qiao Xiao, Boqian Wu, Mykola Pechenizkiy, Decebal Mocanu, and Zhangyang Wang. More convnets in the 2020s: Scaling up kernels beyond 51x51 using sparsity. *arXiv preprint arXiv:2207.03620*, 2022. 1

[13] Yongming Rao, Wenliang Zhao, Yansong Tang, Jie Zhou, Ser-Nam Lim, and Jiwen Lu. Hornet: Efficient high-order spatial interactions with recursive gated convolutions. *arXiv preprint arXiv:2207.14284*, 2022. 1

[14] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11963–11975, 2022. 1, 2, 3, 5, 6

[15] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12104–12113, 2022. 1

[16] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1

[17] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Eur. Conf. Comput. Vis.*, pages 491–507. Springer, 2020. 1

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255, 2009. 1, 3, 4

[19] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10687–10698, 2020. 1

[20] Weijie Su, Xizhou Zhu, Chenxin Tao, Lewei Lu, Bin Li, Gao Huang, Yu Qiao, Xiaogang Wang, Jie Zhou, and Jifeng Dai. Towards all-in-one pre-training via maximizing multi-modal mutual information. *arXiv preprint arXiv:2211.09807*, 2022. 1

[21] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1

[22] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 1

[23] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3558–3568, 2021. 1

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755, 2014. 1, 5

[25] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Int. Conf. Comput. Vis.*, pages 2961–2969, 2017. 1

[26] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: high quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(5):1483–1498, 2019. 1

[27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[28] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 633–641, 2017. 1, 2

[29] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Eur. Conf. Comput. Vis.*, pages 418–434, 2018. 2

[30] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. 2

[31] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 2

[32] Yixuan Wei, Han Hu, Zhenda Xie, Zheng Zhang, Yue Cao, Jianmin Bao, Dong Chen, and Baining Guo. Contrastive learning rivals masked image modeling in fine-tuning via feature distillation. *arXiv preprint arXiv:2205.14141*, 2022. 2

[33] Tingting Liang, Xiaojie Chu, Yudong Liu, Yongtao Wang, Zhi Tang, Wei Chu, Jingdong Chen, and Haibin Ling. Cbnet: A composite backbone network architecture for object detection. *IEEE Trans. Image Process.*, 2022. 2

[34] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Int. Conf. Comput. Vis.*, pages 8430–8439, 2019. 2, 4

[35] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv preprint arXiv:2112.01527*, 2021. 2, 5

[36] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 2, 4, 5

[37] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1209–1218, 2018. 2, 5

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016. 3, 4, 6, 7

[39] Qishuai Diao, Yi Jiang, Bin Wen, Jia Sun, and Zehuan Yuan. Metaformer: A unified meta framework for fine-grained recognition. *arXiv preprint arXiv:2203.02751*, 2022. 3, 4

[40] Jihao Liu, Xin Huang, Yu Liu, and Hongsheng Li. Mixmim: Mixed and masked image modeling for efficient visual representation learning. *arXiv preprint arXiv:2205.13137*, 2022. 4

[41] Mannat Singh, Laura Gustafson, Aaron Adcock, Vinicius de Freitas Reis, Bugra Gedik, Raj Prateek Kosaraju, Dhruv Mahajan, Ross Girshick, Piotr Dollár, and Laurens Van Der Maaten. Revisiting weakly supervised pre-training of visual perception models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 804–814, 2022. 4

[42] Huajun Liu, Fuqiang Liu, Xinyi Fan, and Dong Huang. Polarized self-attention: Towards high-quality pixel-wise regression. *arXiv preprint arXiv:2107.00782*, 2021. 4

[43] Huayao Liu, Jiaming Zhang, Kailun Yang, Xinxin Hu, and Rainer Stiefelhagen. Cmx: Cross-modal fusion for rgb-x semantic segmentation with transformers. *arXiv preprint arXiv:2203.04838*, 2022. 4, 5

[44] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. *arXiv preprint arXiv:2206.05836*, 2022. 4

[45] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. *arXiv preprint arXiv:2211.07636*, 2022. 4

[46] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2918–2928, 2021. 4

[47] Xuan Jin, Wei Su, Rong Zhang, Yuan He, and Hui Xue. Atldetv2. `http://host.robots.ox.ac.uk/leaderboard/displaylb_main.php?challengeid=11&compid=4`, 2019. 4

[48] Yu Liu, Guanglu Song, Yuhang Zang, Yan Gao, Enze Xie, Junjie Yan, Chen Change Loy, and Xiaogang Wang. 1st place solutions for openimage2019–object detection and instance segmentation. *arXiv preprint arXiv:2003.07557*, 2020. 4, 5

[49] Anlin Zheng, Yuang Zhang, Xiangyu Zhang, Xiaojuan Qi, and Jian Sun. Progressive end-to-end object detection in crowded scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 857–866, 2022. 4, 5

[50] Shangliang Xu, Xinxin Wang, Wenyu Lv, Qinyao Chang, Cheng Cui, Kaipeng Deng, Guanzhong Wang, Qingqing Dang, Shengyu Wei, Yuning Du, et al. Pp-yoloe: An evolved version of yolo. *arXiv preprint arXiv:2203.16250*, 2022. 4

[51] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8769–8778, 2018. 3

[52] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *Adv. Neural Inform. Process. Syst.*, 27, 2014. 3

[53] Alejandro López-Cifuentes, Marcos Escudero-Vinolo, Jesús Bescós, and Álvaro García-Martín. Semantic-aware scene recognition. *Pattern Recognition*, 102:107256, 2020. 4

[54] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5356–5364, 2019. 4

[55] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, 2010. 4

[56] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *Int. J. Comput. Vis.*, 128(7):1956–1981, 2020. 4

[57] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 5

[58] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2636–2645, 2020. 5

[59] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 891–898, 2014. 5

[60] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 5

[61] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020. 5

[62] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inform. Process. Syst.*, 34, 2021. 5

[63] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Int. Conf. Comput. Vis.*, pages 4990–4999, 2017. 5

[64] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. *Eur. Conf. Comput. Vis.*, 7576:746–760, 2012. 5

[65] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9308–9316, 2019. 5, 6

[66] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4794–4803, 2022. 5, 6

[67] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 7