Argus: A Compact and Versatile Foundation Model for Vision

Weiming Zhuang[1], Chen Chen[1], Zhizhong Li[1], Sina Sajadmanesh[1], Jingtao Li[1], Jiabo Huang[1], Vikash Sehwag[1], Vivek Sharma[1], Hirotaka Shinozaki[2], Felan Carlo Garcia[2], Yihao Zhan[2], Naohiro Adachi[2], Ryoji Eki[2], Michael Spranger[1], Peter Stone[1,3], Lingjuan Lyu[1*]
[1]Sony AI [2]Sony Semiconductor Solutions [3]University of Texas at Austin

Abstract

While existing vision and multi-modal foundation models can handle multiple computer vision tasks, they often suffer from significant limitations, including huge demand for data and computational resources during training and inconsistent performance across vision tasks at deployment time. To address these challenges, we introduce `Argus`[1]*, a compact and versatile vision foundation model designed to support a wide range of vision tasks through a unified multitask architecture.* `Argus` *employs a two-stage training strategy: (i) multitask pretraining over core vision tasks with a shared backbone that includes a lightweight adapter to inject task-specific inductive biases, and (ii) scalable and efficient adaptation to new tasks by fine-tuning only the task-specific decoders. Extensive evaluations demonstrate that* `Argus`*, despite its relatively compact and training-efficient design of merely 100M backbone parameters (only 13.6% of which are trained using 1.6M images), competes with and even surpasses much larger models. Compared to state-of-the-art foundation models,* `Argus` *not only covers a broader set of vision tasks but also matches or outperforms the models with similar sizes on 12 tasks. We expect that* `Argus` *will accelerate the real-world adoption of vision foundation models in resource-constrained scenarios.*

1. Introduction

Vision foundation models (VFMs) [28, 85] have emerged as powerful models for tackling a broad range of vision tasks, such as image classification, object detection, segmentation, and more. To achieve exceptional transferability, early VFMs [7, 29, 57] focus on learning universal visual feature representations via supervised or self-supervised learning on large-scale datasets, allowing them to generalize well across tasks without requiring extensive customization. However, the substantial costs in task modeling through end-to-end fine-tuning limit the practicality of such representation learning methods in meeting the rapidly growing demands of real-world vision applications [36].

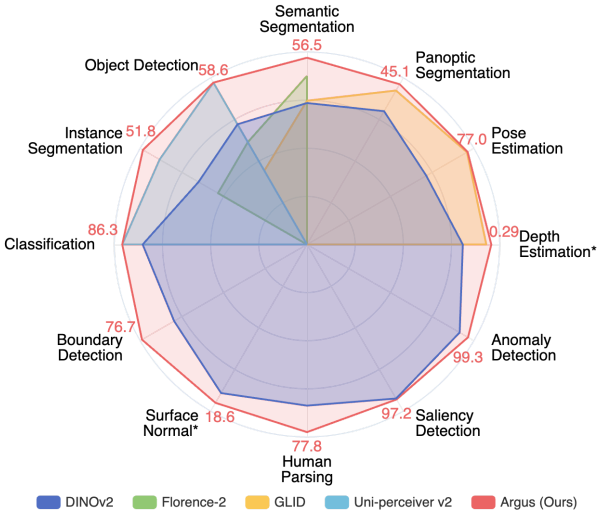


Figure 1. `Argus` achieves strong performance on 12 vision tasks, outperforming existing foundation models such as Florence-2 [75], GLID [43], and Uniperceiver v2 [36]. Adapted DINOv2 [57] refers to DINOv2 plus the same decoders as ours (details in Sec. 3.3). Numbers indicate `Argus`'s performance. * denotes tasks with metrics where the lower is better, for which we reversed the axis direction to align with other tasks for illustration.

Beyond the remarkable success achieved by large language models on individual tasks [6, 19], recent studies of VFMs emphasize *task unification*, which reformulates several vision tasks to share a common input-output structure handled by a unified model [36, 43, 50, 51, 75]. This approach aims to reduce model redundancy and to facilitate task collaboration [58]. Motivated by sequence-to-sequence modeling [9, 10, 68], transformer encoders are used to encode images and other visual modalities into patch tokens, which are then processed and parsed into specific vision outputs by transformer decoders. However, they are not versatile with new tasks, as they require substantial data and

*corresponding author; {weiming.zhuang, lingjuan.lv}@sony.com

[1]The name comes from Argus Panoptes – a hundred-eyed giant with "all-seeing" capability in Greek mythology.

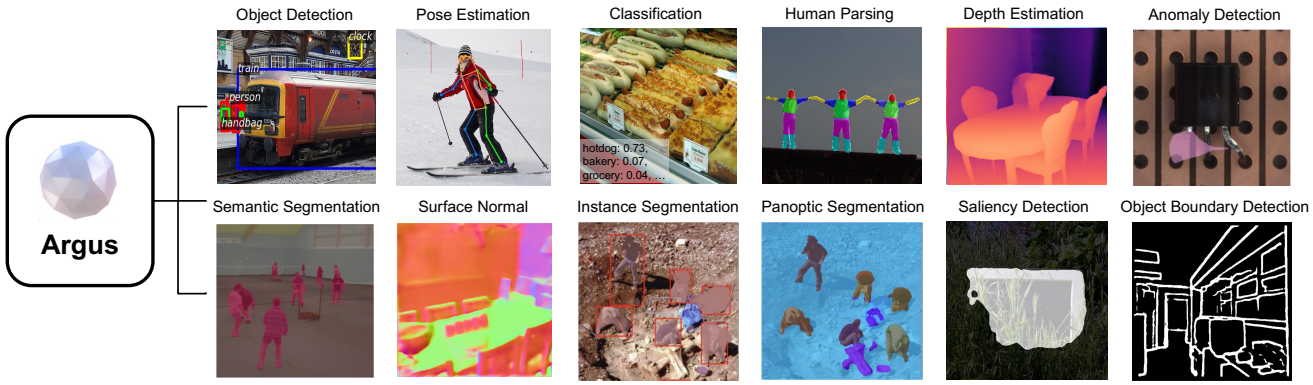


Figure 2. An overview of the 12 computer vision tasks supported by Argus. These figures are the prediction results of Argus. More examples are presented in Appendix E.

computational resources to retrain or fine-tune, due to the lack of task-specific inductive biases in their decoder design [45]. Alternatively, an intuitive yet relatively understudied approach to handling more tasks in VFM is to adopt a multitasking architecture [15, 88]. This technique incorporates multiple tasks' knowledge into one model, potentially enhancing performance across various vision tasks.

In this work, we introduce Argus, a compact and versatile VFM that combines the strengths of the representation learning and task unification paradigms through a multitask architecture, extending the support to a diverse set of vision tasks shown in Fig. 2. To address the challenge of task interference—where tasks compete and hinder the learning of shared structures, we propose a two-stage training strategy (as depicted in Fig. 3):

(i) Multitask Pretraining: Argus is pretrained on a core set of vision tasks, spanning image-level, region-level, and pixel-level perceptions, using a shared backbone and multiple task-specific decoders. Instead of training from scratch, Argus is trained efficiently by leveraging strong representations from a pretrained ViT (*e.g.*, DINOv2 [57]) and only trains a tailored adapter, whose parameters comprise just 13.6% of the backbone. The adapter injects task-specific visual knowledge missing in the pretrained ViT and learns a unified representation that supports diverse tasks.

(ii) Task-Specific Adaptation: this stage fine-tunes only new decoders while keeping the backbone frozen for all tasks. This approach enables Argus to progressively expand its capabilities to new tasks and remain compatible with the advancements in task-specific techniques, such as the latest decoders, simply by appending the decoders to our pretrained backbone, as illustrated in Fig. 3b.

We conduct extensive experiments across 12 vision tasks (Fig. 2) and compare Argus with the latest vision and multi-modal foundation models, including Florence-1&2 [75, 85], Unified-IO 1&2 [50, 51], Uni-Perceiver v2 [36], 4M [53], and more. Argus not only demonstrates a wider task coverage (as shown in Fig. 1), it also delivers stronger performance, achieving the top spot on 10 vision tasks out of 12 and ranking second on object detection and classification. Notably, Argus surpasses methods that use significantly larger models. With around 100M parameters, Argus achieves 56.5% mIoU on the challenging ADE20K semantic segmentation dataset [91, 92], outperforming models like Florence-2 Base with 232M parameters. Compared with huge models, such as Unified-IO XL with 2.9 billion parameters and Unified-IO 2 with 1.1 billion parameters, Argus excels in performance on classification, object detection, and depth estimation. We summarize our main contributions as follows:

- We introduce Argus, a new VFM trained via multitask learning and scalable task-specific adaptation. Argus is highly extensible, allowing efficient adaptation to new tasks by leveraging advanced, task-specific decoders.
- We improve the adapter design that enables ViTs to perform multitask learning effectively, achieving compelling performance while making pretrained ViTs scalable and adaptable to various vision tasks efficiently.
- We conduct extensive experiments on 12 representative computer vision tasks. Argus outperforms existing VFMs and multimodal FMs on most tasks, demonstrating its scalability and effectiveness.

2. Related Work

2.1. Large-scale Foundation Models

The Vision Transformer (ViT) [18] has emerged as a leading model architecture for vision foundation models (VFMs) as it scales effectively with large datasets and computational resources. These models are normally trained on large amounts of data to learn generic feature representations. For example, DINO (DIstillation with NO labels) [7] and DINOv2 [16, 57] learn powerful visual representations through self-supervised learning without any labeled data.

Such generic pretrained models are usually adapted to downstream tasks by fine-tuning into task-specific *specialist models*. This approach is widely adopted in existing VFMs, such as Florence [85], InternImage [73], BEiT [72], Eva[20], and AM-RADIO [60]. For instance, Florence [85] first pretrains on 900M privately curated image-text pairs and then further pretrains its object detection model on a large-scale detection dataset (FLOD-9M). Although such specialist models can achieve strong performance, the costs of adapting them to other tasks and managing distinct backbone parameters for different tasks can escalate significantly as demands for downstream applications grow.

To address these limitations, recent works focus on developing *generalist models* that handle various tasks within a single model. For example, Uni-Perceiver [94] and its successor, Uni-Perceiver v2 [36], use a unified transformer architecture but still lack support for important industrial tasks like depth and pose estimation. Models such as 4M [53], Unified-IO 1&2 [50, 51], and Florence-2 [75] adopt transformer-based encoder-decoder architectures to learn across tasks or modalities, but require vast training datasets (*e.g.*, Unified-IO 2 with 1 billion image-text pairs, Florence-2 with 126M images and 5 billion annotations) and often underperform on key vision tasks like object detection. The recent GLID model [43] addresses adaptability with small linear heads but faces flexibility issues due to its shared transformer decoder, limiting its compatibility with the latest task-optimized decoders.

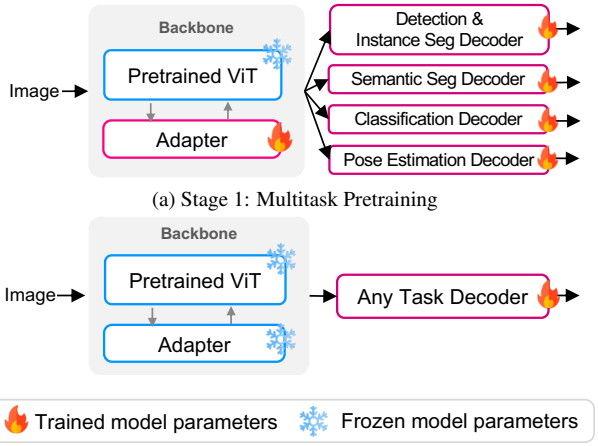
In this work, we aim to offer a compact, training-efficient, scalable and well-performing VFM to better serve various real-world vision applications. Existing VFMs, however, have yet to fully meet these requirements.

2.2. Multitask Learning

Multitask learning (MTL) has become a popular method for generalizing models to handle multiple tasks [8, 15, 69, 71, 88]. By leveraging shared information across tasks, MTL enhances learning through added context and eliminates the need of separate models for different tasks, which results in a more robust and versatile systems [22, 66, 71].

As the main challenge in MTL is managing task interference, where tasks compete for shared resources, hindering optimal learning, research on MTL generally follows two main approaches. The first approach focuses on balancing tasks via loss balancing or gradient balancing. Loss balancing adjusts the contribution of each task to the overall objective function, ensuring that individual task losses are properly weighted [14, 31, 39, 44]. Gradient balancing, on the other hand, equalizes the influence of each task on model updates by adjusting gradients, regardless of differences in task scale or complexity [11, 21, 41, 42, 56, 63, 64, 74, 84].

The second approach focuses on tailoring the network architecture to accommodate diverse tasks while minimizing negative transfer among them [30, 69, 89]. It encompasses two main directions: encoder-focused and decoder-focused techniques. The former customizes the shared backbone structure [5, 24, 25, 44, 52, 80, 81], whereas the latter adopts customized task decoders to address task conflict [4, 54, 67, 70, 78, 80, 82, 93].



(a) Stage 1: Multitask Pretraining

(b) Stage 2: Scalable and Efficient Task-specific Adaptation

Figure 3. Overview of the training process of `Argus`. The multitask pretraining in stage 1 derives a strong backbone, enabling scalable adaptation to new tasks in stage 2 with frozen backbone.

We follow the encoder-focused approach, leveraging a customized shared backbone with multiple standalone task-specific decoders. It is extensible to new tasks and compatible with new advanced task-specific decoders. Built on the robust and general-purpose features of DINOv2 [57], this setup transforms the originally task-agnostic backbone into a task-aware architecture that enhances performance across various downstream tasks, while maintaining flexibility and simplicity in handling diverse objectives.

3. Methodology

We present `Argus`, designed to be compact and efficient in training and capable of performing a variety of vision tasks with a unified multitask architecture. Fig. 3 depicts the two-stage training approach of `Argus`. In the first stage, we pretrain the model using multitask learning across a set of core tasks, with a backbone consisting of a frozen ViT and a lightweight trainable adapter (only 13.6% of the backbone). This approach produces a strong backbone capable of generating high-quality features for diverse vision tasks. In the second stage, we freeze the entire backbone and expand the model's capability to new tasks with scalable task-specific decoder adaptations. In the following, we describe this two-stage training approach in detail.

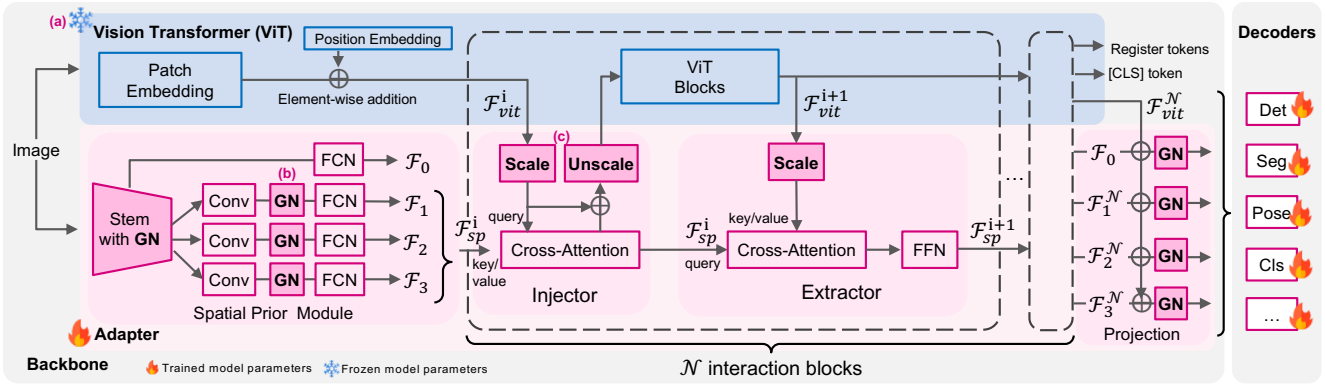


Figure 4. Overview of the Argus model architecture. The backbone consists of a Vision Transformer (ViT) and an adapter. The adapter includes a spatial prior module for spatial feature extraction, injector and extractor modules for interacting features with the ViT blocks, and a projection module for final feature aggregation and normalization. Key recipes for effective multitask learning are highlighted: (a) freezing the pretrained ViT and training only the adapter and decoders in multitask pretraining; (b) replacing all the batch normalization layers with group normalization (GN) layers in the adapter; (c) scaling ViT features before interactions between ViT blocks and the adapter.

3.1. Multitask Pretraining (Stage 1)

Multitask Formulation. Let $T > 1$ be the number of core tasks and $\ell_t(\theta)$ denote the loss of task t given model parameter θ. The goal of multitask pretraining is to minimize the weighted average loss across all core tasks,

$$\theta^* = \arg\min_{\theta} \left\{ \ell_{\text{MTL}}(\theta) := \sum_{t=1}^{T} \lambda_t \, \ell_t(\theta) \right\}, \qquad (1)$$

where λ_t is the weighting factor for task t.

We conduct multitask pretraining across a selected set of core tasks: object detection, instance segmentation, semantic segmentation, pose estimation, and classification. These tasks cover diverse data formats, decoding paradigms, and output densities, spanning image-level classification for capturing high-level semantics (classification), region-level understanding for object and entity localization (object detection and pose estimation), and pixel-level perception for fine-grained image analysis (segmentation). This diversity enables the model to learn robust and versatile features, allowing it to adapt to a broader range of tasks. We ablate the choice of core tasks in Sec. 4.3.

Model. Fig. 4 depicts the model architecture of Argus. The backbone is shared with all the tasks, while each task has its own decoder. The backbone of Argus consists of a ViT initialized with DINOv2 weights [16] and a randomly initialized adapter [12]. Leveraging DINOv2's powerful representations for both image-level and pixel-level tasks, the adapter enhances the capability by introducing vision-specific inductive biases such as the multiscale features that strengthen performance in region-based tasks, *e.g.*, object detection, as shown in Fig. 1.

Our adapter design enhances ViT-Adapter [12], with modifications tailored specifically for MTL. The adapter contains a convolutional spatial prior module (SPM) that introduces multiscale features, along with $\mathcal{N}$ interaction blocks, each containing an injector and an extractor to interact the multiscale adapter features bidirectionally with the single-scale ViT features $\mathcal{F}_{vit}$ via cross-attentions. To be specific, the multiscale spatial feature $\mathcal{F}_{sp}$ is formed by flattening and concatenating the outputs $\{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3\}$ of SPM, which have resolutions at $\frac{1}{8}$, $\frac{1}{16}$, and $\frac{1}{32}$ of the input image size, respectively. In addition, SPM also produces $\mathcal{F}_0$ at $\frac{1}{4}$ resolution, which is sent to output without participating in interaction blocks. The ViT's output patch tokens, with the resolution $\frac{1}{16}$, are added to the multiscale adapter features $\{\mathcal{F}_0, \mathcal{F}_1^{\mathcal{N}}, \mathcal{F}_2^{\mathcal{N}}, \mathcal{F}_3^{\mathcal{N}}\}$ after spatial interpolation. This approach fully leverages the strong DINOv2 features, requiring the adapter to learn only the residuals necessary to enhance the feature quality. We provide more details about the adapter in Appendix A.1.

Recipes for Improved Adapter Design in MTL. The original ViT-Adapter [12] struggles in MTL setting, suffering from the negative transfer across tasks. To address this issue, we introduce several key architectural modifications that improve the performance as follows.

(a) Freezing the ViT. Unlike the standard practice in ViT-Adapter that trains all parameters, we empirically found that freezing the ViT accelerates convergence and improves generalization. By retaining the pretrained DINOv2 weights, we prevent catastrophic forgetting—the loss of previously learned knowledge [32]—and significantly reduce the trainable parameters (down to 13.6%). Thence the model can leverage the full representational power of DINOv2 during training, rather than using it merely as initialization.

(b) Replacing Batch Normalization (BN) With Group Normalization (GN). We observed that using BN led to suboptimal performance, as different tasks share the same set of BN statistics during inference, but they are conflicting across tasks. To resolve this, we replace all BN layers in the adapter with GN layers, leading to consistent performance improvements across all tasks.

(c) Scaling ViT Features Before Interaction. The token norms in ViT increase progressively after each layer, causing the adapter to interact with tokens of varying norm scales. To counter this, we scale the ViT tokens per image before interaction by a factor of $\sqrt{d}/\overline{\|x\|_2}$, where d is the feature dimension and $\overline{\|x\|_2}$ is the average L2 norm of tokens. After passing through the injector module, we unscale the tokens to their original scale. This approach enables the adapter to interact with tokens that have a stable norm.

3.2. Scalable Task-Specific Adaptation (Stage 2)

Following the multitask pretraining, we efficiently extend `Argus` to various vision tasks by freezing the entire backbone and only fine-tuning task-specific decoders. All tasks leverage the shared backbone while using their individual decoders. Unlike other FMs that either produce a specialist model after fine-tuning [85] or require extensive retraining to cover new tasks [36, 50, 51, 94], `Argus` naturally evolves as a versatile model, easily adaptable to additional tasks by adding new decoders.

In this work, we scale `Argus` to support 12 important vision tasks, as shown in Fig. 2. For tasks requiring multiscale features, like object detection, the decoders forward these features from the backbone directly. For tasks that do not need multiscale features, such as classification, we flatten and concatenate the features from the backbone before passing them to the decoder. We then train the task-specific decoders using their corresponding loss functions with backbone frozen.

3.3. Implementation Details

Training Procedure of MTL. Our MTL implements a *multi-input* paradigm [38], where each task utilizes its own dedicated dataset. The total loss is computed as a weighted sum of task-specific losses, where each loss is scaled by its task weight. To improve training efficiency, we developed a new MTL framework and implemented automatic mixed-precision training, enabling all core MTL tasks to fit within a single GPU's memory. This optimization allows us to leverage Distributed Data Parallel (DDP) for multi-GPU training. More details can be found in Appendix A.2.

Preserving the ViT Structure. ViT-Adapter [12] modifies the ViT structure by replacing self-attention with window attention [1] to reduce time and memory. Instead, we retain the original ViT structure as we aim to preserve the DINOv2's pretrained knowledge. To improve efficiency without altering the architecture, we leverage the efficient attention library from xFormers [34]. Furthermore, ViT-Adapter changes DINOv2's patch embedding from 14×14 to 16×16 to align with the adapter's patch size. Instead of modifying the embedding weights, we rescale the image resolution by $^{14}/_{16}$. Finally, we keep the class token and register tokens intact while ViT-Adapter discards them.

Task-Specific Decoders. In the multitask pretraining stage, we use the Mask DINO [35] as the decoder for object detection and instance segmentation, an attentional pooler [83] followed by a linear layer for classification, UperNet [76] for semantic segmentation, and DPT [59] for heatmap regression of pose keypoints. In the task-specific adaptation, we use Mask2Former [13] for instance and panoptic segmentation; UperNet [76] for human parsing, object boundary detection and surface normal; DPT [59] for saliency detection and depth estimation. The detailed description of these tasks are provided in Appendix B.

Model Configuration. By default, we use the ViT-Base and adapter with $\mathcal{N} = 4$ interaction blocks as the backbone, which has ∼100M parameters in total, including 13.6M trainable parameters of the adapter. Multitask pretraining on 5 core tasks for 300K iterations takes ∼4 days on 8 NVIDIA H100 GPUs. Subsequent task-specific adaptations to the other 7 tasks can be completed within 12 hours under the same hardware setting. We provide ablations with a larger backbone size in Sec. 4.3.

4. Experiments

In this section, we begin with the experiment setup, and then compare the performance of `Argus` with other FMs. We ablate design choices and discuss the results subsequently.

4.1. Experiment Setup

Datasets. In the multitask pretraining stage, we use the COCO dataset [40] for object detection, instance segmentation, and pose estimation, the ADE20K dataset [91, 92] for semantic segmentation, and the ImageNet dataset [17] for classification. In the task-specific adaptation stage, we extend to panoptic segmentation using the ADE20K dataset [92]. Depth estimation, object boundary detection, and surface normal estimation are adapted using the NYUv2 [65] dataset. PASCAL-Context [55] is used for human parsing and saliency detection. Anomaly detection is adapted using a random subset of MVTecAD [2]: transistor, metal nut, screw, and leather (referred to as MVTecAD-4). We use the standard splits for model training and evaluation, which are entirely distinct without data leakage. Summary statistics of these datasets are presented in Appendix C.

Methods	# Params	Semantic Segmentation		Object Detection	Instance Segmentation		Panoptic Segmentation	Image Classification
		ADE20K mIoU ↑	NYUv2 mIoU ↑	COCO AP_b ↑	COCO AP_m ↑	ADE20K AP_m ↑	ADE20K PQ ↑	ImageNet Top-1 Acc. ↑
MAE-B [28]	86M	46.1	-	48.3	39.9	-	-	84.2
4M-B [53]	86M	50.1	-	49.7	42.7	-	-	84.2
DINOv2 (ViT-B) [57]	86M	52.5	61.4	55.1	47.8	31.4	43.4	84.6
InterImage-B [73]	97M	51.3	-	50.3	44.8	-	-	84.9
GLID (Swin-B) [43]	126M	52.7	-	51.2	-	30.9	44.7	-
Florence [85]	893M	-	-	**62.4**	-	-	-	**90.0**
Florence-2-B* [75]	232M	-	-	41.4	-	-	-	-
Florence-2-B (fine-tuned) [75]	232M	54.9	-	53.6	46.4	-	-	-
Uni-perceiver v2-B* [36]	308M	-	-	58.6	50.6	-	-	86.3
Unified-IO XL* [50]	2.9B	-	-	-	-	-	-	79.1
Unified-IO 2* [51]	1.1B	-	-	47.2	-	-	-	-
Argus	100M	**56.5**	**64.7**	58.6	**51.8**	**37.5**	**45.1**	86.3

Table 1. Performance comparison of Argus with recent FMs over 5 vision tasks on COCO [40], ADE20K [91, 92], NYUD-v2 [65], and ImageNet [17] datasets. "-" means the model did not cover the task. "*" denotes the models with a unified decoder for multiple tasks. Best results are in bold, the second-best results are underlined. Unlike other FMs that only cover few tasks, Argus supports all the vision tasks in the table. Note that for fair comparison, we only compare to foundation models that can support multiple vision tasks covered in Fig. 2.

Methods	# Params	Pose Estimation		Depth Estimation	Boundary Detection	Surface Normal	Human Parsing	Saliency Detection	Anomaly Detection
		COCO		NYUv2	NYUv2	NYUv2	PASCAL-C	PASCAL-C	MVTecAD-4
		AP_k ↑	AR ↑	RMSE ↓	odsF ↑	mErr ↓	mIoU ↑	maxF ↑	I-AUROC ↑
DINOv2 (ViT-B) [57]	86M	57.2	62.2	0.307	71.5	19.3	76.7	97.1	98.3
GLID (Swin-B) [43]	126M	76.7	-	0.293	-	-	-	-	-
Unified-IO XL* [50]	2.9B	-	-	0.385	-	-	-	-	-
Unified-IO 2* [51]	1.1B	-	-	0.423	-	-	-	-	-
Argus	100M	**77.0**	**81.8**	**0.290**	**76.7**	**18.6**	**77.8**	**97.2**	**99.3**

Table 2. Continuation of Tab. 1: Performance comparison of Argus with recent FMs over the other 7 vision tasks on COCO [40], NYUv2 [33], PASCAL-C [55] and MVTecAD-4 [2] datasets. We exclude methods listed in Tab. 1 that do not support these tasks.

In total, we use approximately 1.6 million images to train Argus.

Metrics. For object detection, we use COCO with mean Average Precision (AP_b) across IoU thresholds. Pose estimation on COCO uses Average Precision (AP_k) and Average Recall (AR). Semantic segmentation is evaluated on ADE20K and NYUv2 using mean Intersection over Union (mIoU). Panoptic segmentation on ADE20K uses Panoptic Quality scores (PQ). Instance segmentation on COCO and ADE20K use mean Average Precision (AP_m). Depth estimation, object boundary detection, and surface normal prediction are assessed on NYUv2 with Root Mean Square Error (RMSE), optimal-dataset-scale F-measure (odsF), and mean angular error (mErr), respectively. Human parsing and saliency detection on PASCAL-Context uses mIoU and maximal F-measure (maxF). Classification is measured by Top-1 accuracy (Top-1) on ImageNet [17]. Anomaly detection is evaluated on MVTecAD-4 with Area Under the Receiver Operating Characteristic Curve (I-AUROC). More experiment setups are shown in Appendix A.4.

4.2. Performance Comparison

We compare the performance of Argus with both the state-of-the-art VFMs and multi-modal FMs that support multiple vision tasks covered in Fig. 2, including Florence [85] and Florence-2 [75], Uni-perceiver v2 [36], 4M [53], MAE [28], Unified-IO [50] and Unified-IO 2 [51], DINOv2 [57], and GLID [43]. Since DINOv2 itself does not support any tasks, its results are obtained by training the same decoders as in Argus (refer to Sec. 3.3). We also provide extra comparison with some MTL models and task-specific models in Appendix D.3.

Tabs. 1 and 2 compare the performance of Argus against the aforementioned baselines across 12 vision tasks on the datasets listed in Sec. 4.1. These results demon-

Freeze ViT	Norm	Scale	EMA	Iter.	COCO AP_b	COCO AP_m	COCO AP_k	ADE20K mIoU	ImageNet Top-1
×	GN	✓	×	200k	55.3	49.0	74.9	51.3	82.8
✓	BN	✓	×	200k	56.7	50.2	74.3	55.0	84.7
✓	GN	×	×	200k	57.8	51.1	74.7	55.5	85.2
✓	GN	✓	×	200k	57.7	51.1	75.2	55.6	85.1
✓	GN	✓	✓	200k	57.9	51.3	75.6	**56.9**	85.0
✓	GN	✓	✓	300k	**58.6**	**51.8**	**77.0**	56.5	**86.3**

Table 3. Ablation studies on multitask pretraining recipe. Based on row 4, we modify the ingredients of *(a) freeze ViT*, *(b) norm*, and *(c) scale*, each shown in the first three rows. In the last two rows, we add the *EMA* and increase the training iterations. Key modifications in each row are highlighted in blue.

MTL Algorithm on 5 Tasks	COCO AP_b	COCO AP_m	COCO AP_k	ADE20K mIoU	ImageNet Top-1
FAMO [42]	57.4	50.9	75.0	**55.6**	84.6
GradNorm [11]	57.2	50.9	74.6	55.0	**85.9**
Empirical Task Weighting	**57.7**	**51.1**	**75.2**	**55.6**	85.1

Table 4. Comparison of empirical task weighting with MTL algorithms, including the loss-balancing algorithm FAMO [42] and the gradient-balancing algorithm GradNorm [11].

strate that Argus is the most versatile VFM for perception tasks, support all the 12 vision tasks, whereas others only support a limited subset. Argus is also among the top-performing VFM, surpassing comparable models on 10 out of 12 tasks. Florence [85] achieves higher performance in object detection and image classification, but it has approximately 8× more parameters and is trained on a significantly larger detection dataset. Notably, Argus achieves 56.5% mIoU on ADE20K semantic segmentation with only 100M parameters, while Florence-2-B [75] achieves 54.9% with 232M parameters—more than double the size of Argus. Compared with Uni-perceiver v2, Argus matches its performance in object detection and image classification and surpasses it on instance segmentation. Argus also consistently outperforms DINOv2 across all tasks. In addition to the aforementioned results, we discuss the fairness of the comparison in Appendix D and complement our quantitative analysis with qualitative results in Appendix E.

4.3. Ablation Studies

Multitask Pretraining Recipe. Tab. 3 shows the effects of the ingredients in our MTL training recipe in Sec. 3.1. The key observations are as follows. (a) Comparing row 1 to row 4 reveals that freezing ViT is crucial, as training the entire model does not converge well (validation curves in Appendix D.5). (b) Comparing row 2 to row 4 shows that replacing BN with GN in the adapter boosts performance on all tasks and alleviates negative transfer issues in MTL (validation curves in Appendix D.5). (c) Comparing row 3 to row 4 indicates that scaling ViT features before interaction improves the pose estimation from 74.7% to 75.2% while maintaining the performance of other tasks.

Based on the configurations in row 4, we further improve the overall performance by keeping an exponential moving average (EMA) of the weights (row 5 *vs*. row 4) and increasing the training iterations (row 6 *vs*. row 5). Finally, the setting in row 6 is adopted in Argus.

Multitask Learning Algorithm. Tab. 4 compares the representative loss-balancing algorithm FAMO [42] and the gradient-balancing algorithm GradNorm [11] with our naive empirical task weighting. Detailed configurations of these MTL algorithms and our task weights are provided in Appendix A.3. The results show that FAMO has the lowest classification top-1 accuracy, while GradNorm un-

MTL Tasks	COCO AP_b	COCO AP_m	COCO AP_k	ADE20K mIoU	ImageNet Top-1
All 5 Tasks	57.7	51.1	75.2	55.6	85.1
w/o Det. & Instance Seg.	54.2	48.1	75.1	55.6	85.3
w/o Pose Estimation	57.5	51.0	74.4	55.5	85.2
w/o Semantic Seg.	57.9	51.1	75.6	54.2	85.2
w/o Classification	57.8	50.9	75.6	55.6	83.3

Table 5. Multitask pretraining with fewer tasks. Tasks marked in blue means the tasks removed from MTL pretraining (stage 1 in Fig. 3) and only covered in the task-specific adaptation (stage 2).

Models	Mask R-CNN AP_b	Mask R-CNN AP_m	Mask DINO AP_b	Mask DINO AP_m	CO-DETR AP_b
DINOv2 [57]	19.2	13.3	55.1	47.8	56.6
Argus	**53.0**	**47.3**	**58.8**	**52.0**	**60.5**

Table 6. Performance of Argus using different decoders for object detection and instance segmentation on the COCO dataset.

Models	# Params	COCO AP_b	COCO AP_m	COCO AP_k	ADE20K mIoU	ImageNet Top-1
Argus-Base	100M	58.6	51.8	77.0	55.6	86.3
Argus-Large	327M	60.2	53.1	78.0	58.6	87.1

Table 7. Ablation of backbone sizes. Increasing backbone size leads to better performance across all tasks.

Models	Person Cam. Sec. $AP_{50:95}$	Person Cam. Sec. AP_{50}	Pascal Person $AP_{50:95}$	Pascal Person AP_{50}	Det. Cars $AP_{50:95}$	Det. Cars AP_{50}
4M-B [53]	18.1	34.5	28.7	43.7	51.8	56.6
Florence-2-B [75]	43.8	73.7	39.1	51.6	39.5	42.4
Argus	**56.6**	**97.7**	**68.5**	**83.7**	**79.1**	**81.8**

Table 8. Comparison of object detection performance on unseen data. $AP_{50:95}$ and AP_{50} are used as the evaluation metric.

Freeze Adapter	COCO AP_b	COCO AP_m	COCO AP_k	ImageNet Top-1	ADE20K mIoU	ADE20K AP_m	ADE20K PQ	NYUv2 RMSE ↓	NYUv2 odsF	NYUv2 mIoU	PASCAL-Context mErr ↓	PASCAL-Context mIoU	PASCAL-Context maxF
✓	58.6	51.8	77.0	86.3	56.5	37.5	45.1	0.290	75.8	64.7	18.6	77.8	97.2
×	**58.8**	**52.0**	**77.4**	**86.4**	**56.7**	**37.6**	**45.7**	**0.281**	**76.5**	**65.3**	**17.6**	**79.0**	**97.7**

Table 9. Performance of Argus with and without freezing the adapter in the task-specific adaptation (stage 2 in Fig. 3).

derperforms on pose estimation and semantic segmentation. In contrast, our empirical task weighting achieves the best performance on four out of five tasks. We hypothesize that the combination of drastically different task types and the multi-input training paradigm presents a unique challenge for existing MTL algorithms. We leave the systematic study of MTL algorithms for future work.

Choosing Core Tasks. We conduct ablation studies on the core task selection for multitask pretraining in Tabs. 5 and 13. In Tab. 5, we find that excluding one task can lead to minor improvement (under 0.5%) on some of the other tasks, likely due to reduced task interference. However, the excluded task experiences a substantial drop in performance when finetuning the decoder with a frozen backbone. We also examine an alternative set of 5 tasks (row 4 of Tab. 13), but observe lower performance on almost all tasks, presumably due to limitations in task diversity and data quantity. Additionally, we experiment with expanding core tasks to 8 and 11 in Tab. 13 in the Appendix. While our MTL framework can support diverse task combinations, these expanded tasks often lead to overfitting on the additional tasks due to their limited data. Although training losses decrease over the course of training, the validation performance of these new tasks decline in later iterations (Appendix D.5). These results indicate that our current 5 core tasks in pretraining strikes a good balance between the effective multitask pretraining and new task adaptations.

Decoder Flexibility. Unlike models with a single shared decoder [36, 43, 75], Argus can easily support new task-specific decoders. For example, as shown in Tab. 6, Argus can leverage high-performing decoders like CO-DETR [95] for higher performance or opt for traditional decoders like Mask R-CNN [27] in resource-constrained scenarios. This flexibility in Argus is beneficial in real-world applications where specialized decoders are preferred for specific scenarios, such as Iter-Deformable-DETR [90] for crowded scenes and SCRFD [23] for face detection.

Backbone Sizes. We conduct an ablation with a larger backbone, Argus-Large, using ViT-Large with 303M parameters and a slightly larger adapter with 23.7M parameters. Tab. 7 shows that increasing the backbone size leads to performance improvements across all tasks, highlighting the potential of scaling the backbone for further performance improvement.

Generalization on the Unseen Datasets. We further compare the performance of Argus with 4M [53] and Florence-2 [75] on unseen data. We use Pascal VOC 2012 Person, Person Camera Security and Detecting Cars datasets from Roboflow Universe for evaluation, where all models are not trained on these datasets. We evaluate performance using two metrics: average precision calculated over multiple IoUs thresholds ranging from 0.5 to 0.95 ($AP_{50:95}$), and average precision at IoU threshold 0.5 (AP_{50}). Tab. 8 shows that Argus achieves substantially better performance. We provide dataset sources and evaluate on more unseen public datasets for image classification in Appendix D.4.

Fine-tuning Adapter in Task-specific Adaptation. We perform an ablation study to examine the impact of adapter during task-specific adaptation. As shown in Tab. 9, fine-tuning the adapter yields consistent performance gains across all tasks. While these improvements are relatively modest—under 0.5% for core tasks and under 1.5% for other tasks—they highlight the strong representational capability of the multitask-pretrained adapter in effectively transferring to diverse vision tasks.

5. Conclusion

In this work, we introduce Argus, a compact, well-performing, training-efficient and scalable vision foundation model (VFM) for a diverse range of vision tasks. Extensive experiment results on 12 vision tasks demonstrate that Argus achieves compelling performance compared to existing VFMs or multi-modal FMs. Moving forward, our model's inherent flexibility opens up exciting possibilities for more vision tasks such as OCR, low-level vision tasks, *etc*. We envision that our VFM and proposed approaches can accelerate the practical, real-world adaptation of VFMs in resource-constrained applications.

Acknowledgment We thank Nidham Gazagnadou and Xin Dong for their input during early stages of this work.

References

[1] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 5

[2] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 5, 6, 15

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4009–4018, 2021. 3

[4] Deblina Bhattacharjee, Tong Zhang, Sabine Süsstrunk, and Mathieu Salzmann. Mult: An end-to-end multitask learning transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12031–12041, 2022. 3

[5] Deblina Bhattacharjee, Sabine Süsstrunk, and Mathieu Salzmann. Vision transformer adapters for generalizable multitask learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19015–19026, 2023. 3

[6] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 1

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1, 2

[8] Rich Caruana. Multitask learning. *Machine learning*, 28: 41–75, 1997. 3

[9] Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*, 2021. 1

[10] Ting Chen, Saurabh Saxena, Lala Li, Tsung-Yi Lin, David J Fleet, and Geoffrey E Hinton. A unified sequence interface for vision tasks. *Advances in Neural Information Processing Systems*, 35:31333–31346, 2022. 1

[11] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018. 3, 7, 1, 2

[12] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. In *The Eleventh International Conference on Learning Representations*, 2023. 4, 5

[13] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 5, 3

[14] Sumanth Chennupati, Ganesh Sistu, Senthil Yogamani, and Samir A Rawashdeh. Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 3

[15] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020. 2, 3

[16] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 4

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 6

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[19] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360*, 2021. 1

[20] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19358–19369, 2023. 3

[21] Heshan Fernando, Han Shen, Miao Liu, Subhajit Chaudhury, Keerthiram Murugesan, and Tianyi Chen. Mitigating gradient bias in multi-objective learning: A provably convergent approach. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[22] Golnaz Ghiasi, Barret Zoph, Ekin D Cubuk, Quoc V Le, and Tsung-Yi Lin. Multi-task self-training for learning general representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8856–8865, 2021. 3

[23] Jia Guo, Jiankang Deng, Alexandros Lattas, and Stefanos Zafeiriou. Sample and computation redistribution for efficient face detection. *arXiv preprint arXiv:2105.04714*, 2021. 8

[24] Pengsheng Guo, Chen-Yu Lee, and Daniel Ulbricht. Learning to branch for multi-task learning. In *International conference on machine learning*, pages 3854–3863. PMLR, 2020. 3

[25] Hussein Hazimeh, Zhe Zhao, Aakanksha Chowdhery, Maheswaran Sathiamoorthy, Yihua Chen, Rahul Mazumder, Lichan Hong, and Ed Chi. Dselect-k: Differentiable selection in the mixture of experts with applications to multi-task learning. In *Advances in Neural Information Processing Systems*, pages 29335–29347. Curran Associates, Inc., 2021. 3

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 8, 2

[28] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 1, 6

[29] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 1

[30] Zhuoliang Kang, Kristen Grauman, and Fei Sha. Learning with whom to share in multi-task feature learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 521–528, 2011. 3

[31] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018. 3

[32] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 4

[33] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2661–2671, 2019. 6, 5

[34] Benjamin Lefaudeux, Francisco Massa, Diana Liskovich, Wenhan Xiong, Vittorio Caggiano, Sean Naren, Min Xu, Jieru Hu, Marta Tintore, Susan Zhang, Patrick Labatut, Daniel Haziza, Luca Wehrstedt, Jeremy Reizenstein, and Grigory Sizov. xformers: A modular and hackable transformer modelling library. https://github.com/facebookresearch/xformers, 2022. 5

[35] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3041–3050, 2023. 5, 2

[36] Hao Li, Jinguo Zhu, Xiaohu Jiang, Xizhou Zhu, Hongsheng Li, Chun Yuan, Xiaohua Wang, Yu Qiao, Xiaogang Wang, Wenhai Wang, et al. Uni-perceiver v2: A generalist model for large-scale vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2691–2700, 2023. 1, 2, 3, 5, 6, 8

[37] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018. 3

[38] Baijiong Lin and Yu Zhang. LibMTL: A Python library for multi-task learning. *Journal of Machine Learning Research*, 24(209):1–7, 2023. 5

[39] Baijiong Lin, Feiyang Ye, Yu Zhang, and Ivor W Tsang. Reasonable effectiveness of random weighting: A litmus test for multi-task learning. *arXiv preprint arXiv:2111.10603*, 2021. 3

[40] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 5, 6, 7, 13

[41] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021. 3

[42] Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. FAMO: Fast adaptive multitask optimization. In *Neural Information Processing Systems Foundation*, 2023. 3, 7, 1, 2

[43] Jihao Liu, Jinliang Zheng, Yu Liu, and Hongsheng Li. Glid: Pre-training a generalist encoder-decoder vision model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22851–22860, 2024. 1, 3, 6, 8

[44] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1871–1880, 2019. 3

[45] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 2

[46] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20402–20411, 2023. 3

[47] Jonathan Long, Evan Shelhamer, Trevor Darrell, and UC Berkeley. Fully convolutional networks for semantic segmentation. arxiv 2015. *arXiv preprint arXiv:1411.4038*, 2014. 3

[48] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[49] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 2

[50] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. In *The Eleventh International Conference on Learning Representations*, 2022. 1, 2, 3, 5, 6

[51] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision, language, audio, and action. *arXiv preprint arXiv:2312.17172*, 2023. 1, 2, 3, 5, 6

[52] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1930–1939, 2018. 3

[53] David Mizrahi, Roman Bachmann, Oguzhan Kar, Teresa Yeo, Mingfei Gao, Afshin Dehghan, and Amir Zamir. 4m: Massively multimodal masked modeling. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 6, 7, 8, 5, 13

[54] Eslam Mohamed and Ahmad El Sallab. Spatio-temporal multi-task learning transformer for joint moving object detection and segmentation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 1470–1475. IEEE, 2021. 3

[55] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014. 5, 6, 7, 21, 22

[56] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. *arXiv preprint arXiv:2202.01017*, 2022. 3

[57] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 2, 3, 6, 7, 5, 12, 16, 18, 19, 20, 21, 22, 23

[58] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1

[59] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 5, 2, 3

[60] Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Am-radio: Agglomerative vision foundation model reduce all domains into one. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12490–12500, 2024. 3

[61] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 12

[62] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 5, 12

[63] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018. 3

[64] Dmitry Senushkin, Nikolay Patakin, Arseny Kuznetsov, and Anton Konushin. Independent component alignment for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20083–20093, 2023. 3

[65] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012. 5, 6, 7, 16, 17, 23

[66] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *International conference on machine learning*, pages 9120–9132. PMLR, 2020. 3

[67] Benyuan Sun, Jin Dai, Zihao Liang, Congying Liu, Yi Yang, and Bo Bai. Gppf: A general perception pre-training framework via sparsely activated multi-task learning. *arXiv preprint arXiv:2208.02148*, 2022. 3

[68] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112, Cambridge, MA, USA, 2014. MIT Press. 1

[69] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? *Advances in neural information processing systems*, 8, 1995. 3

[70] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 527–543. Springer, 2020. 3

[71] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3614–3633, 2021. 3

[72] 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. 3

[73] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14408–14419, 2023. 3, 6

[74] Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. *arXiv preprint arXiv:2010.05874*, 2020. 3

[75] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a vari-

ety of vision tasks. *arXiv preprint arXiv:2311.06242*, 2023. 1, 2, 3, 6, 7, 8, 5, 13

[76] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. 5, 3

[77] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 2, 3, 5, 6, 14

[78] Hanrong Ye and Dan Xu. Inverted pyramid multi-task transformer for dense scene understanding. In *European Conference on Computer Vision*, pages 514–530. Springer, 2022. 3

[79] Hanrong Ye and Dan Xu. Invpt: Inverted pyramid multi-task transformer for dense scene understanding. *arXiv preprint arXiv:2203.07997*, 2022. 3, 5, 6, 7, 16, 17

[80] Hanrong Ye and Dan Xu. Taskprompter: Spatial-channel multi-task prompting for dense scene understanding. In *ICLR*, 2023. 3

[81] Hanrong Ye and Dan Xu. Taskexpert: Dynamically assembling multi-task representations with memorial mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21828–21837, 2023. 3, 5

[82] Hanrong Ye and Dan Xu. Invpt++: Inverted pyramid multi-task transformer for visual scene understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 3

[83] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 5, 2

[84] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020. 3

[85] 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, 2, 3, 5, 6, 7

[86] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draema discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8330–8339, 2021. 3, 5, 6, 15

[87] Jin Zeng, Yanfeng Tong, Yunmu Huang, Qiong Yan, Wenxiu Sun, Jing Chen, and Yongtian Wang. Deep surface normal estimation with hierarchical rgb-d fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6153–6162, 2019. 3

[88] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34 (12):5586–5609, 2021. 2, 3

[89] Xiangyun Zhao, Haoxiang Li, Xiaohui Shen, Xiaodan Liang, and Ying Wu. A modulation module for multi-task learning with applications in image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 401–416, 2018. 3

[90] Anlin Zheng, Yuang Zhang, Xiangyu Zhang, Xiaojuan Qi, and Jian Sun. Progressive end-to-end object detection in crowded scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 857–866, 2022. 8

[91] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 5, 6

[92] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 2, 5, 6

[93] Ling Zhou, Zhen Cui, Chunyan Xu, Zhenyu Zhang, Chaoqun Wang, Tong Zhang, and Jian Yang. Pattern-structure diffusion for multi-task learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4514–4523, 2020. 3

[94] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16804–16815, 2022. 3, 5

[95] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6748–6758, 2023. 8