This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

LoftUp: Learning a Coordinate-Based Feature Upsampler for Vision Foundation Models

Haiwen Huang[1,2] Anpei Chen[1,2] Volodymyr Havrylov[1]
Andreas Geiger[1,2] Dan Zhang[3]
[1]University of Tübingen [2] Tübingen AI Center [3] Bosch Center for Artificial Intelligence

Abstract

Vision foundation models (VFMs) such as DINOv2 and CLIP have achieved impressive results on various downstream tasks, but their limited feature resolution hampers performance in applications requiring pixel-level understanding. Feature upsampling offers a promising direction to address this challenge. In this work, we identify two critical factors for enhancing feature upsampling: the upsampler architecture and the training objective. For the upsampler architecture, we introduce a coordinate-based cross-attention transformer that integrates the high-resolution images with coordinates and low-resolution VFM features to generate sharp, high-quality features. For the training objective, we propose constructing high-resolution pseudo-groundtruth features by leveraging class-agnostic masks and self-distillation. Our approach effectively captures fine-grained details and adapts flexibly to various input and feature resolutions. Through experiments, we demonstrate that our approach significantly outperforms existing feature upsampling techniques across various downstream tasks. Our code is released at https://github.com/andrehuang/loftup.

1. Introduction

High-quality pretrained representations from Vision Foundation Models (VFMs) have become standard for a wide range of computer vision tasks [10, 16, 21, 37, 41, 43, 45, 54, 56, 62, 63]. However, because of the patrification or aggressive pooling operations in VFMs, the output features are typically 16 or even more times smaller in spatial resolution than the input images, limiting their utility for tasks that require fine-grained, pixel-level understanding.

To address the challenge of limited feature resolution in VFMs, one straightforward approach is to use larger image inputs to obtain higher-resolution features. However, processing high-resolution inputs incurs a quadratic increase in computational cost and can introduce severe artifacts if the VFMs are not trained for such resolutions. Moreover,

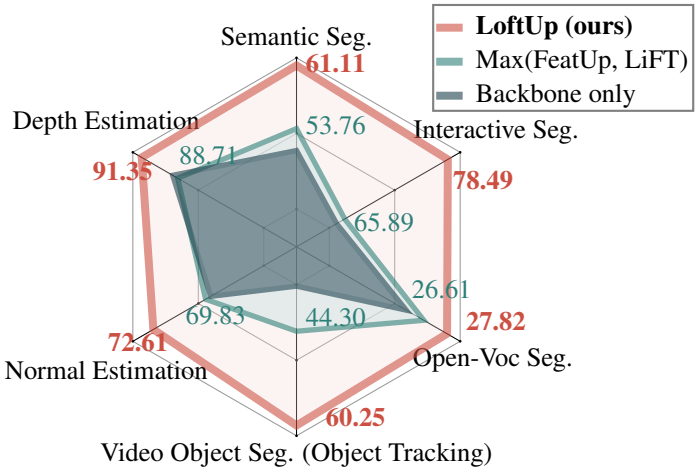
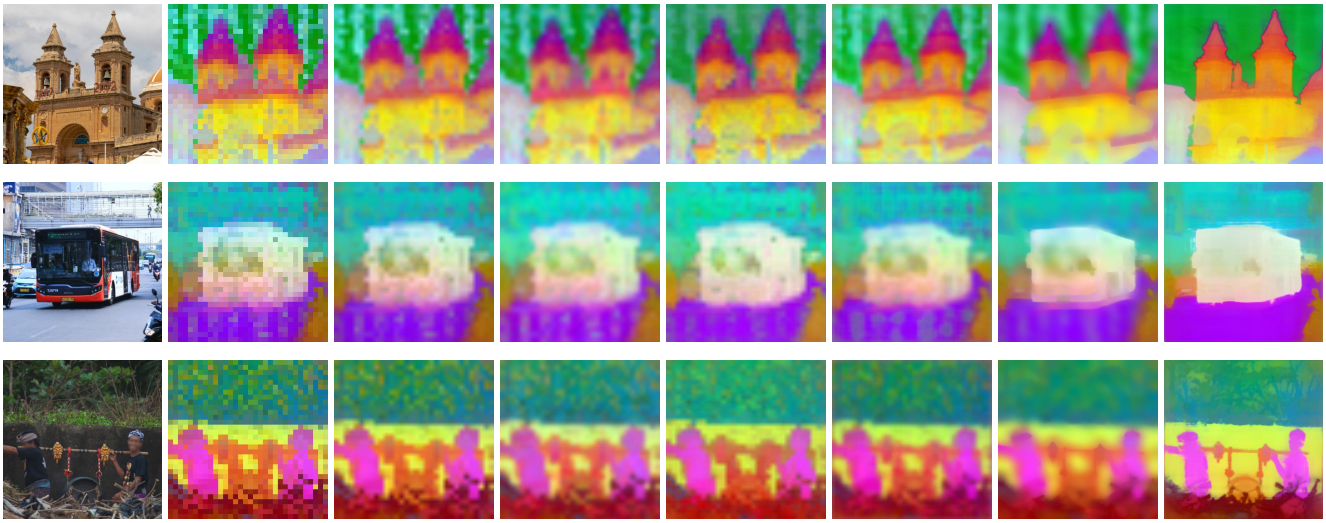


Figure 1. **LoftUp improves significantly across various tasks over the VFM backbone (DINOv2-S [37]) and current SoTA feature upsampling performance (FeatUp [12] and LiFT [52]).** See experiment details in Sec. 5.

training or fine-tuning VFMs on high-resolution images demands substantial computational resources and meticulous tuning [37, 43, 45, 55]. An alternative strategy is to train task-specific decoders that leverage multi-layer intermediate features to upsample VFM outputs [3, 16, 24, 27, 62]. Yet, this approach often comes with significant training costs and necessitates retraining the decoder for each new application. In addition, many downstream tasks lack sufficient data needed to fine-tune a high-quality decoder.

More recently, FeatUp [12] and LiFT [52] have independently introduced task-agnostic feature upsamplers trained with general reconstruction losses, demonstrating that such methods can substantially enhance VFM performance across a variety of tasks. By upsampling VFM features and pairing them with a lightweight task-specific decoder, these approaches provide a promising alternative that avoids heavy, task-specific training while achieving a more generalizable solution. Nonetheless, as shown in Fig. 1, current feature upsamplers still fall short of reaching optimal performance.



(a) Original image (b) Low res (c) Bilinear (d) Resize-conv (e) LIIF [2] (f) LiFT [52] (g) FeatUp [12] (h) **LoftUp** (Ours)

Figure 2. **Comparison of features from upsamplers.** Backbone is DINOv2-S/14 [37].

In this work, we systematically explore the design space of feature upsamplers and identify two critical components: the upsampler architecture and the training objective. The architecture determines the capacity of the upsampler to learn effectively, while the training objective defines the upper performance limit. By optimizing both elements, our approach achieves substantially stronger results than previous state-of-the-art methods.

Regarding the **upsampler architecture**, to capture high-resolution details while avoiding the cumulative artifacts from multiple layers of interpolation or deconvolution, we propose a simple coordinate-based transformer that directly predicts high-resolution features for each pixel. Specifically, our model takes image coordinates and RGB values as inputs and performs cross-attention with the low-resolution VFM feature map. This facilitates a fine-grained, content-aware mapping from coordinates to high-resolution features, effectively bypassing the constraints imposed by fixed local kernels and standard upsampling layers. Moreover, unlike the implicit approach in FeatUp [12], which requires test-time optimization, our method learns feature upsampling directly from the training dataset and generalizes to diverse scenes without additional test-time adjustments.

For the **training objective**, the primary challenge is the absence of groundtruth high-resolution feature annotations. Although downstream task labels such as depth or masks can be used, this approach risks compromising the task-agnostic nature of the upsampler and hinders its ability to generalize to unseen tasks. Due to this challenge, previous task-agnostic feature upsampling works, FeatUp [12] and LiFT [52], both compute the training loss at a low resolution. In this work, we address these limitations by constructing pseudo-groundtruth (pseudo-GT) features directly at the input image resolution. Specifically, we leverage class-agnostic masks generated by off-the-shelf segmentation foundation models [21, 45] to ensure that the pseudo-GT accurately reflects the underlying geometry and delineates object boundaries. We further refine the pseudo-GT using a self-distillation strategy that reduces noise and artifacts. This high-resolution pseudo-GT enables loss computation at a high resolution, empowering the upsampler to learn fine-grained details.

In Fig. 2, we qualitatively show that LoftUp yields markedly sharper and more detailed features than alternative upsampling methods. To further demonstrate the versatility and effectiveness of our approach, we evaluate it on a range of downstream tasks such as semantic segmentation, depth estimation, and video object segmentation. As shown in Fig. 1, our approach leads to performance gains of 10–20% over previous SoTA upsamplers for most tasks, and an impressive nearly 50% improvement on video object segmentation [39]. Furthermore, thanks to its coordinate-based design, our upsampler adapts seamlessly to various input and feature resolutions, catering to the diverse requirements of downstream applications. Overall, with less than a 20% increase in parameters compared to the original foundation models, our feature upsampler offers a task-agnostic, lightweight, and plug-and-play enhancement that significantly boosts VFM backbones across multiple tasks.

2. Related Work

Feature upsampling refers to increasing the spatial resolution of a feature map. In this work, our goal is to increase the feature resolution to the original image resolution, that is, full resolution. Traditional non-learnable methods include various ways of interpolation [7, 33] and image-

adaptive filtering such as joint bilateral filtering (JBU) [22] and guided image filtering [15]. In modern deep learning, previous work has proposed various architecture- and downstream-task-specific feature upsamplers. For example, Index Networks [29] and A2U [5] are effective on image matting, but fall short in other tasks. PointRend [20] proposes a point-rendering method specifically for upsampling segmentation output. And CARAFE [57], SAPA [31], and FADE [30] are proposed specifically for encoder-decoder architectures. More recently, with the success of vision foundation models such as DINOv2 [37] and CLIP [41], there is a trend for feature upsamplers to be downstream-task-agnostic so that they can be used with the VFM backbone together in various applications [12, 52]. Our work falls into this category. With this task-agnostic goal in mind, we further explain the architecture and training objective design as follows.

Architecture for Feature Upsamplers. Traditional upsampler architectures rely on multiple layers of interpolation or deconvolution to transform low-resolution features into higher resolutions. Examples include JBU [22], standard deconvolution [8, 35, 49], resize-convolution [36] and U-Net-style upsampling modules [47, 52]. However, the multi-layer design inevitably leads to error accumulation, resulting in increased blurriness as the resolution increases.

In this work, inspired by coordinate-based methods in 3D reconstruction [25, 53, 60, 61], we adopt a coordinate-based approach and view feature upsampling as a mapping from high-resolution coordinates to high-resolution features. This effectively bypasses the limitations of standard upsampling layers. Previously, FeatUp also proposed a coordinate-based network (MLP) for feature upsampling [12]. However, their approach requires per-image optimization and is therefore not scalable. Another related work, LIIF [2], also employs an MLP to parameterize high-resolution outputs but is limited to local feature interactions. In contrast, our method does not need test-time optimization and enables global interactions between image inputs and low-resolution features through a cross-attention mechanism, leading to stronger upsampling performance.

Task-agnostic Training Objective for Feature Upsampling. Due to the absence of groundtruth high-resolution features, it is a challenge to create high-quality training objective for task-agnostic feature upsampling. FeatUp [12] and LiFT [52] were the first two works to propose a task-agnostic training pipeline for feature upsamplers. However, their training is at a low resolution, leaving the high-resolution features severely under-constrained. We will explain them in more detail Sec. 4. In this paper, we propose a self-distillation approach to generate full-resolution pseudo-GT, which is then used to supervise feature upsampling training at full resolution. This approach fully unlocks the potential of the feature upsampler to capture fine-grained details in high-resolution images.

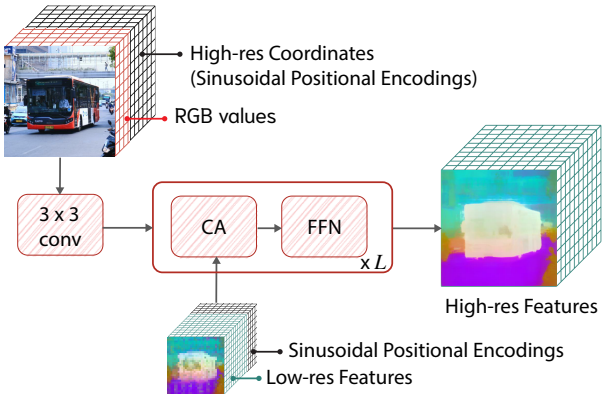


Figure 3. **Architecture of LoftUp.** Our coordinate-based network with cross-attention mechanism effectively integrates the fine-grained details from image RGB values and semantically-rich low-res features to produce high-resolution feature maps.

3. Coordinate-Based Feature Representation

Prior works typically address feature upsampling as a gradual resolution enhancement process, where features are progressively lifted to a higher resolution. In contrast, we adopt a coordinate-based representation of the high-resolution features and view feature upsampling as a mapping from a pixel coordinate (x, y) to the high-resolution features at that pixel. Specifically, we propose a cross-attention mechanism to effectively incorporate the low-resolution features and the high-resolution image inputs with coordinates.

As shown in Fig. 3, following prior work on 3D coordinate-based reconstruction [34, 61], our model encodes full-resolution coordinates with sinusoidal positional embeddings and concatenates them with RGB values. A convolutional layer then projects this combined input into the feature dimension. Next, an L-block cross-attention transformer uses these high-resolution features as queries and low-resolution VFM features as keys and values, producing the final high-resolution feature map.

Our cross-attention design enables high-frequency details to interact globally with semantically rich representations, facilitating global-content-aware upsampling. This overcomes the limitations of fixed or locally predicted kernels in prior feature upsamplers. As illsustrated in Fig. 2, multi-layer upsamplers such as resize-conv [36], LiFT (U-Net) [52], and FeatUp (a modified JBU) [12] tend to produce blurry outputs with artifacts. Additionally, while LIIF [2] employs a coordinate-based design, its reliance on local interactions limits its upsampling quality. In contrast, our LoftUp accurately captures object boundaries and produces fine-grained feature maps with minimal artifacts.

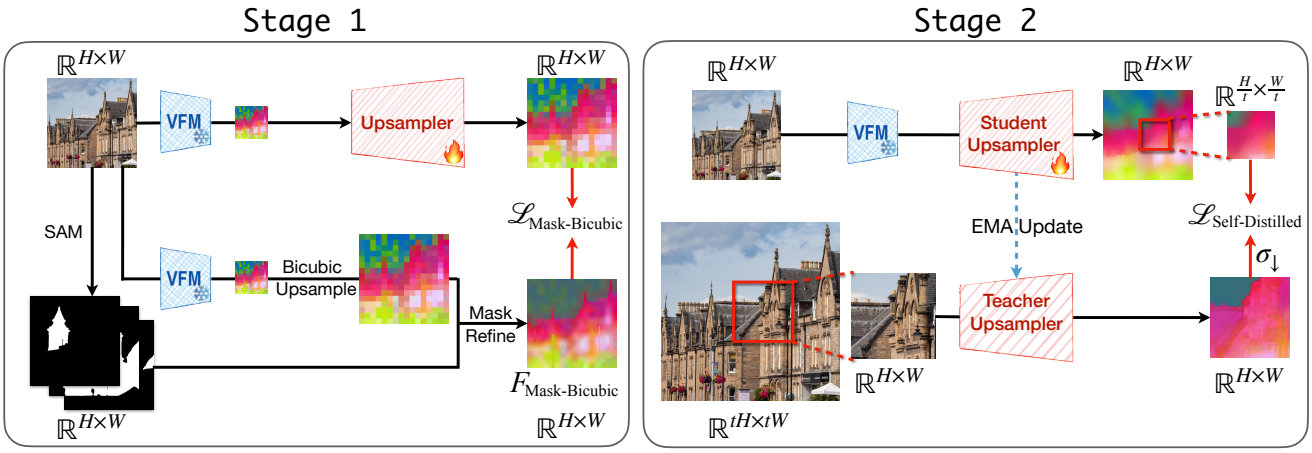


Figure 4. **Our two-stage LoftUp training approach.** Stage 1 trains an upsampler with class-agnostic masks to refine bicubic-upsampled features. Stage 2 employs self-distillation, initializing teacher and student upsamplers from Stage 1's pre-trained model. All VFM image inputs share the resolution ($H \times W$). For visual clarity, the VFM block is omitted from Stage 2's teacher branch.

Furthermore, our coordinate-based approach allows flexible generation of feature maps at arbitrary resolutions by adjusting the input coordinate resolution—unlike traditional upsampling methods, which are restricted to fixed scaling factors due to their multi-layer structures. Finally, we note that while attention mechanisms can be computationally expensive, our cross-attention remains relatively efficient because it processes a much smaller set of low-resolution tokens as keys and values, rather than the progressively higher-resolution feature maps as in multi-layer upsampler architectures. In fact, as our experiments (Tab. 6) later show, our upsampler's inference speed is comparable to bilinear upsampling.

4. Training Objective

While the architecture of the upsampler establishes its capacity, the training objective ultimately sets the performance ceiling. A central challenge for training a task-agnostic feature upsampler is the absence of full-resolution groundtruth features. As a result, previous work has either relied on proxy tasks or constructed pseudo-GT at lower resolutions. For instance, FeatUp [12] employs a multi-view reconstruction task: the predicted high-resolution features, $\hat{F}_{\text{HR}}$, are first transformed via an affine mapping t, then downsampled, and finally compared against the low-resolution features extracted from images that go through the same transformation, i.e.,

$$\mathcal{L}_{\text{FeatUp}} = \mathcal{D}\big(f(t(I)), \sigma_{\downarrow}(t(\hat{F}_{\text{HR}})\big).$$

In contrast, LiFT [52] directly uses features from $2\times$ larger images $I_{2\times}$ as pseudo-GT, constraining the upsampler to predict $2\times$ upsampled features $\hat{F}_{2\times}$ via

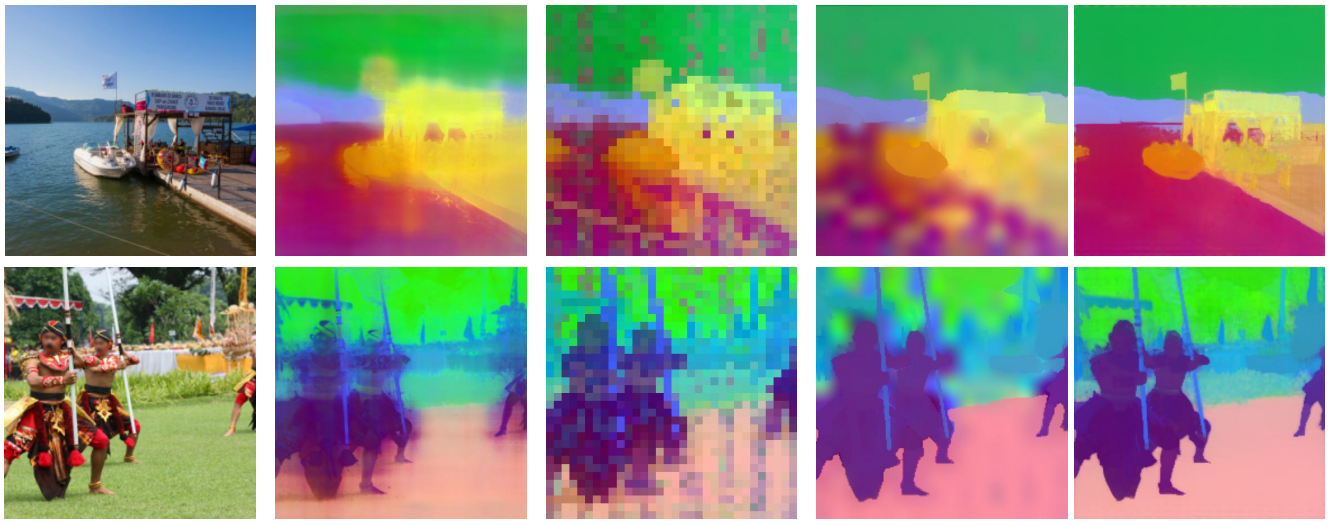
$$\mathcal{L}_{\text{LiFT}} = \mathcal{D}\big(\hat{F}_{2\times}, f(I_{2\times})\big).$$

However, since both $\mathcal{L}_{\text{FeatUp}}$ and $\mathcal{L}_{\text{LiFT}}$ are at relatively low resolutions (only 1/16 or 1/8 of the target resolution), they provide only weak supervisory signals for capturing fine-grained details inherent in high-resolution outputs, potentially leaving the upsampled features under-constrained.

In this work, we employ a self-distillation strategy to generate high-quality pseudo-GT for supervising full-resolution features, as shown in Fig. 4. First, we train an upsampler using high-resolution class-agnostic masks, which emphasize sharp boundaries and geometric awareness. In the second stage, we enhance training through self-distillation. Specifically, we first initialize a teacher and a student upsampler both from the trained upsampler in Stage 1. Then, the teacher processes high-resolution image crops, and its outputs serve as supervision for the corresponding student upsampler outputs. By distilling more detailed and accurate feature maps from the teacher, Stage 2 further reduces noise and enhances sharpness in the student's outputs. Overall, by constructing high-quality pseudo-GT feature maps at full-resolution, our approach enables the upsampler to capture more detailed structures and boundaries, ultimately pushing the limits of feature upsampling.

4.1. Stage 1: Training with class-agnostic masks

Low-resolution features and their upsampled features using existing upsamplers are often too blurry and noisy to be used as pseudo-GT. In contrast, Segment Anything Model (SAM) [21] produces full-resolution, class-agnostic masks that capture fine-grained details such as small object parts and boundaries. Previous work has shown that training with these masks leads to strong task generalization [21, 44–46]. These qualities make them well suited for refining feature maps to reduce artifacts and noise while promoting smoothness in homogeneous regions.



(a) Original image (b) Per-image optimized (c) 2x features (LiFT) (d) Mask-Bicubic (Stage 1) (e) Self-Distilled (Stage 2)

Figure 5. **Visualization of different pseudo-GT.** Both Mask-Bicubic and Self-Distilled are proposed by our work. We set $\alpha = 0.8$ (in Eq. (1)) to balance sharp boundaries from masks and fine-grained details from high-res features.

To leverage these masks, we first upsample a low-resolution feature map to full resolution via bicubic interpolation, yielding F_{Bicubic}. Then, for each mask $m \in M = \{m_1, m_2, ..., m_N\}$, we compute the mean feature, $\overline{F_{\text{Bicubic}}[m]}$, and blend it with the original features, yielding a *mask-refined* feature map at pixels within mask m:

$$F_{\text{Mask-Bicubic}}[m] = \alpha * \overline{F_{\text{Bicubic}}[m]} + (1-\alpha) * F_{\text{Bicubic}}[m], \quad (1)$$

where $\alpha \in [0, 1]$ controls the degree of mask refinement.

We then supervise the upsampler using the loss:

$$\mathcal{L}_{\text{Mask-Bicubic}} = ||\hat{F}_{\text{HR}} - F_{\text{Mask-Bicubic}}||_2, \quad (2)$$

which encourages high-resolution outputs with homogeneous features within each mask region, thereby leveraging the rich structural information from the class-agnostic masks. Note while these masks can also refine features from more complicated upsampling methods such as FeatUp [12] or JBU [22], we observe that simple bicubic upsampling already yields strong results, making it the preferred choice for Stage 1 training.

4.2. Stage 2: Training with self-distillation

We further enhance our upsampler training using self-distillation. To distill high-quality features, we adopt a teacher-student dual-branch design. Both the teacher and student upsamplers are initialized with the upsampler trained in Stage 1. The teacher is updated via an Exponential Moving Average (EMA) during training, ensuring a stable and progressively improving target.

For the student branch, each image is resized to a fixed resolution $I \in \mathbb{R}^{H \times W}$ (224px or 336px), matching the VFM's input requirements. For the teacher branch, the original image is resized to a larger size $I_{\text{HR}} \in \mathbb{R}^{tH \times tW}$ with

Metrics	Depth map	Per-image implicit	2x features (LiFT)	Mask-Bicubic	Self-Distilled
High-Res Loss	✓	✓	✗	✓	✓
Task-Agnostic	✗	✓	✓	✓	✓
Geometry Fidelity	✓	✓	✗	✓	✓
Free of Noise/Artifacts	✓	✗	✗	✗	✓
Scalability	✓	✗	✓	✓	✓
COCO Seg. (IoU)	56.02	59.08	52.90	59.87	**61.11**
Cityscapes Seg. (IoU)	44.82	46.05	35.85	50.43	**53.10**

Table 1. **Comparison of different pseudo-GT choices.**

$t \in [2, 4]$. The models then process on a crop from I_{HR} that also matches the VFM's resolution, i.e., $\text{crop}(I_{\text{HR}}) \in \mathbb{R}^{H \times W}$. Note our original images, sourced from the SA1B dataset [21], have a minimum resolution of 1500px on the shortest side and is thus larger than $tH \times tW$. While a higher t yields more fine-grained teacher outputs, it also lowers the supervision resolution on the student ($H/t \times W/t$), reducing supervision effectiveness. In practice, we find $t \in [2, 4]$ strikes a good balance. The teacher's output on this crop, $f_{\text{teacher}}(\text{crop}(I_{\text{HR}})) \in \mathbb{R}^{H \times W}$, is subsequently downsampled via $\sigma_{\downarrow}(\cdot)$ to match the resolution of the corresponding student output, $\text{crop}(f_{\text{student}}(I)) \in \mathbb{R}^{H/t \times W/t}$, and used as pseudo GT for the student. Formally, the self-distillation loss is defined as:

$$\mathcal{L}_{\text{Self-Distilled}} = \mathcal{D}\Big(\sigma_{\downarrow}\big(f_{\text{teacher}}(\text{crop}(I_{\text{HR}}))\big), \text{crop}\big(f_{\text{student}}(I)\big)\Big),$$

where $\mathcal{D}$ denotes a discrepancy function between features. In our experiments, we adopt the affinity matrix loss [40, 58, 59], which consistently outperforms the standard L_2 loss. This self-distillation strategy leverages the teacher's abil-

Method	Semantic Seg.		Depth Estimation		Normal Estimation	
	COCO mIoU ↑	CS mIoU ↑	RMSE ↓	Recall ↑	RMSE ↓	Recall ↑
Low-res	51.21	36.54	<u>0.1071</u>	<u>89.08</u>	32.29	69.56
Bilinear	56.15	<u>44.79</u>	0.1132	87.68	<u>32.27</u>	<u>70.03</u>
LiFT	53.35	35.80	0.1078	88.71	32.31	69.78
FeatUp	<u>56.30</u>	44.19	0.1092	88.57	32.25	69.83
LoftUp	**61.11**	**53.10**	**0.0921**	**91.35**	**30.79**	**72.76**

Table 2. **Comparison of feature upsamplers across tasks of semantics segmentation, depth estimation, and normal estimation.** For each metric, we boldface the best performance and underline the second best.

Method	Video Obj. Seg.			Open-Voc Seg.			Interactive Seg.		
	J Mean	F Mean	J & F Mean	COCO [26]	CS [4]	ADE [64]	GrabCut [48]	Berkeley [32]	DAVIS [38]
Low-res	42.05	31.27	36.66	25.70	34.48	19.50	64.90	55.77	52.82
Bilinear	42.62	33.90	38.26	25.78	34.56	19.53	65.04	55.83	54.26
LiFT	<u>47.68</u>	36.78	42.23	25.96	<u>35.39</u>	19.50	29.66	31.99	41.14
FeatUp	45.70	<u>42.90</u>	<u>44.30</u>	<u>26.61</u>	35.01	<u>19.99</u>	<u>65.89</u>	<u>56.67</u>	<u>55.03</u>
LoftUp	**58.72**	**61.79**	**60.25**	**27.82**	**38.82**	**21.29**	**78.49**	**65.24**	**67.31**

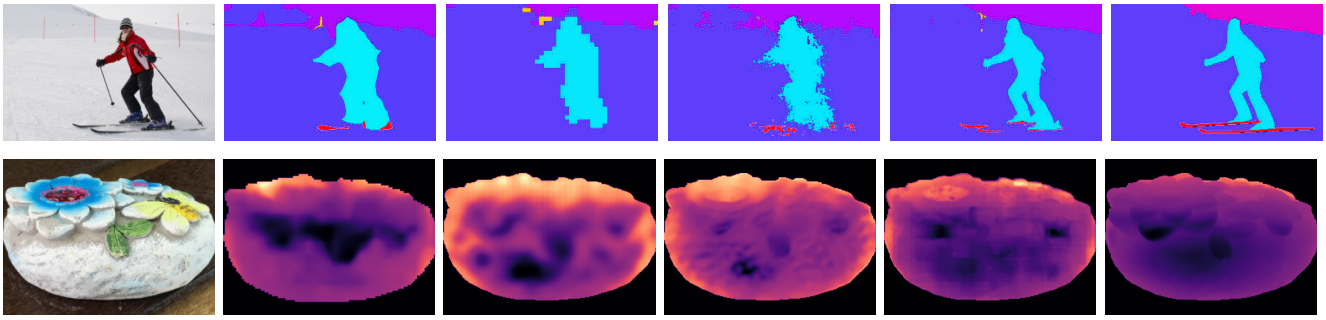
Table 3. **Comparison of feature upsamplers across tasks of video object segmentation, zero-shot open-vocabulary segmentation, and interactive segmentation.** Following previous works [16, 17, 21, 23, 27, 52], we report J Mean, F Mean, and their average for video object segmentation, mIoU for open-vocabulary segementation, and IoU@1 Click for interactive segmentation.

ity to handle an easier task—generating high-quality features for cropped regions—thus providing a more reliable pseudo-GT to guide the student. Furthermore, to capitalize on the sharp boundary priors provided by the class-agnostic masks, we apply the same mask refinement described in Eq. (1) to the teacher's features. For simplicity, we denote this refined output as $F_{\text{Self-Distilled}}$. As shown in Fig. 5, $F_{\text{Self-Distilled}}$ reduces the blurry artifacts in $F_{\text{Mask-Bicubic}}$ and captures the underlying geometry much better.

Further Discussion on Pseudo-GT Choices. Earlier in this section, we identified low-resolution loss as a bottleneck in prior training objectives. Here, we expand on that discussion by addressing additional key factors uncovered during our pseudo-GT exploration. In particular, Tab. 1 compares our approach with three alternative pseudo-GT choices: (1) depth maps, (2) per-image optimized high-resolution features from FeatUp-Implicit [12], and (3) features from 2× larger inputs, as in LiFT [52]. First, pseudo-GT should be **task-agnostic** to ensure generalizability across downstream tasks. For example, although depth maps contain fine-grained geometric details, they lack semantic information, potentially limiting the upsampler's ability to maintain global semantic consistency. Furthermore, pseudo-GT must **capture image geometry accurately**, maintain sharp boundaries and clear color transitions, and **minimize artifacts and noise**. As shown in Fig. 5, per-image optimized features exhibit halo artifacts around objects and speckle noise in smooth regions, while 2× features introduces significant mosaic artifacts. $F_{\text{Mask-Bicubic}}$ still suffers from blurriness and blobby artifacts inherited from bicubic upsampling. Among these, only $F_{\text{Self-Distilled}}$ meets these requirements, demonstrating the effectiveness of our self-distillation framework. Finally, **scalability** is another crucial factor. Per-image optimization is computationally expensive, requiring 1 minute per HR feature map on an A100 GPU and 74MB for storage —equating to 35 GPU days and 3.5TB storage for just 50K images. In contrast, other methods generate pseudo-GT in real time with a single forward pass, taking less than 0.1 seconds per image. Overall, we find that $F_{\text{Self-Distilled}}$ best satisfies these requirements and delivers the strongest performance.

5. Experiments

In this section, we compare LoftUp with other upsampling approaches on various tasks and conduct in-depth analysis on the strengths of LoftUp. By default, we use the DINOv2-S/14 model [37] as VFM. Experiments with CLIP [41] and RADIO [43] show that DINOv2 produces the best downstream performance. Detailed results for CLIP and RADIO are provided in the supplements. Following previous works [12, 52], we resize input images to a resolution of 224 and upsample the VFM features to match the input resolution (14× for DINOv2-S/14). All upsamplers are trained on a 1M-image subset of the SA1B dataset [21] and additional implementation details are available in the supplements.



(a) Original image (b) Bilinear (c) LiFT (d) FeatUp (e) **LoftUp** (f) Groundtruth

Figure 6. **Visualization of predictions examples** on semantic segmentation on COCO-Stuff [1] and depth estimation on NAVI [18]. We provide additional visualization in the supplements.

5.1. Cross-Task Comparison

We compare LoftUp with two most recent task-agnostic feature upsamplers: FeatUp (JBU) [12] and LiFT [52], along with bilinear upsampling as a baseline. As demonstrated in Tab. 2 and Tab. 3, LoftUp consistently outperforms all alternatives on six downstream tasks.

Since we expect high-resolution features to *enhance fine-grained scene understanding*, we first evaluate them on **semantic segmentation** following [12, 13]. We train a linear projection on upsampled features to predict coarse classes in COCO-Stuff [1] and Cityscapes [4]. LoftUp achieves 7.3% and 15.6% relative improvements over previous best methods. As expected, Cityscapes benefits more from feature upsampling due to its high number of small objects.

To assess the upsamplers' *generalizability to geometry prediction*, we follow [10] and evaluate **depth and normal estimation** using a lightweight DPT decoder head [42] on the NAVI dataset [18]. Among the compared methods, LoftUp is the *only* upsampling method that significantly outperforms the low-res baseline, achieving 2.5% and 4.4% relative improvements in recall.

We further evaluate the *temporal consistency* of upsampled features through **video object segmentation** on DAVIS2017 [39], which requires consistent object tracking across video frames. Following prior works [17, 52], we use feature affinity maps to track objects across frames and report J Mean (mIoU, reflecting region similarity), F Mean (mean F-score, reflecting contour accuracy) and their average. LoftUp achieves significant performance improvements, with a 39.6% increase in J Mean and a 97.6% increase in F Mean over the low-resolution baseline, demonstrating a strong ability to identify object boundaries.

Finally, to assess the *flexibility of usage across modalities* like text and click embeddings, we evaluate on **zero-shot open-vocabulary segmentation** with ProxyCLIP [23] where DINOv2 features adjust CLIP features spatially for better vision-language alignment, and on **interactive segmentation** using a modified SimpleClick [27] where visual features are incorporated with click embeddings for segmentation. Results show that LoftUp is the *only* method to achieve consistent, significant improvements over the low-res and bilinear baselines, demonstrating its adaptability to multimodal alignment.

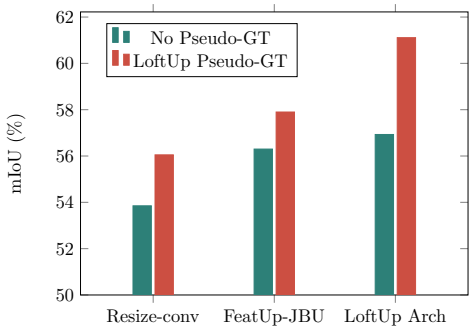


Figure 7. **LoftUp pseudo-GT improves different upsamplers.** Here we report semantic segmentation results on COCO.

Architecture	COCO	Cityscapes	Depth Rec.	Normal Rec.
Low-res	51.21	36.54	89.08	69.56
Resize-conv	56.05	45.94	88.38	65.59
LIIF	52.24	40.36	79.85	46.06
FeatUp-JBU	57.90	44.83	89.38	68.83
LoftUp	**61.11**	**53.10**	**91.35**	**72.61**

Table 4. **Architecture comparison** on semantic segmentation and depth and normal estimation. All upsamplers are trained using LoftUp training objective.

5.2. More Analysis of LoftUp Design

Our pseudo-GT provides strong training signal. In Fig. 5 and Tab. 1, we already discuss and compare different pseudo-GT choices both qualitatively and quantitatively by training LoftUp architecture using different pseudo-GTs. In Fig. 7, we further show that our pseudo-GT can be used to enhance other upsamplers such as resize-conv and FeatUp (JBU), demonstrating broad applicability of the strong, full-resolution supervision signal of our pseudo-GT alone.

Our architecture significantly outperforms other alternatives when training under the same LoftUp training ob-

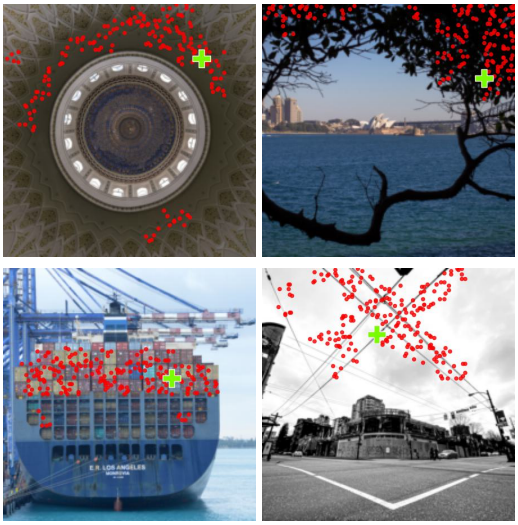


Figure 8. **Visualization of attended regions** (shown as dots) in the low-resolution features corresponding to a high-resolution pixel (marked as a cross). The density of dots represents the values in the attention map. LoftUp leverages relevant information from the global feature map to upsample features at each pixel.

jective as in Sec. 4. As shown in Tab. 4, we compare LoftUp with resize-conv, LIIF, and FeatUp-JBU. We demonstrate that while other methods may bias towards certain tasks, LoftUp achieves the strongest performance across all of the tasks, indicating its stronger capacity to leverage the high-quality pseudo-GT. We further provide ablation studies of our architecture in Tab. B.3 in supplements.

LoftUp benefits from the long-range, content-aware image-feature interaction thanks to the cross-attention mechanism. Instead of using a local kernel as in most previous feature upsampling works, our cross-attention mechanism effectively allows global attention and content-aware upsampling. In Fig. 8, we visualize the attended region of a pixel in high-res coordinate (in cross) in the corresponding low-resolution features (in dots) where the density of dots reflects the values of attention maps. As shown, the attended regions often exhibit semantically similar patterns across the image. Consequently, our upsampler leverages relevant information from the entire image, enhancing global semantic consistency and refining details at object boundaries.

LoftUp supports arbitrary upsampling scales via its coordinate-based design. As shown in Tab. 5, LoftUp maintains strong performance across varying input and feature resolutions. Here, "input res" refers to the image resolution fed into the VFM, while "feature res" denotes the final feature map size generated by LoftUp. Notably, LoftUp outperforms the original VFM backbone even when the latter processes images at 2× resolution—an approach that incurs nearly 4× the computational and memory cost. This highlights LoftUp's efficiency and adaptability, making it well-suited for diverse scaling needs in downstream tasks.

Method	Input res	Feature res	COCO	Cityscapes
Low-res	224	16	51.21	36.54
2x-large input	448	32	55.42	50.29
4x-large input	896	64	56.79	58.45
LoftUp	224	56	59.87	48.20
		112	60.75	51.04
		224	**61.11**	**53.10**
		448	60.61	52.29
	448	56	60.16	51.57
		112	61.50	56.46
		224	61.78	58.55
		448	**62.26**	**61.24**

Table 5. **LoftUp maintains strong performance with diverse input sizes and upsampling scales.**

Architecture	COCO	Cityscapes	Params (M)	Infer time (s/img)
Bilinear	56.15	44.79	22.1	0.0604
Resize-conv	56.05	45.94	27.5 (+5.4)	0.0922
LIIF	52.24	40.36	24.0 (+1.9)	0.1757
LiFT	53.35	35.80	23.3 (+1.2)	**0.0773**
FeatUp-JBU	57.90	44.83	**22.3 (+0.2)**	0.1213
FeatUp-Implicit	51.12	44.81	22.5 (+0.4)	54.302
LoftUp	**61.11**	**53.10**	26.4 (+4.3)	0.0893

Table 6. **Comparison of efficiency for different upsampler architectures.** Inference time is computed on a single A100 GPU.

LoftUp also enjoys relatively high efficiency. In Tab. 6, we compare the parameter size and inference time of each model. Interestingly, while LoftUp has slightly more parameters than most alternatives (still less than 20% of the VFM), it is the second fastest upsampler after LiFT, which only does 2x upsampling. This efficiency arises because LoftUp directly maps coordinates to features, bypassing multiple intermediate upsampling layers used by other methods. Moreover, compared to the coordinate-based FeatUp-Implicit that requires per-image optimization, LoftUp is both significantly faster and more effective.

6. Conclusion

In this work, we systematically explored upsampler architectures and training objectives for feature upsampling and introduced LoftUp, a coordinate-based cross-attention transformer upsampler trained with our proposed self-distilled high-resolution pseudo-GT. Our experiments demonstrate that LoftUp generates sharp, fine-grained feature maps, enables global content-aware image-feature interaction, supports arbitrary upsampling scales, and achieves fast inference. Notably, LoftUp outperforms existing methods across six diverse downstream tasks. We hope our work inspires further research in feature upsampling and contributes to the vision community by enhancing VFMs for a wide range of downstream applications.

Acknowledgement

Haiwen Huang and Dan Zhang were supported by Bosch Industry on Campus Lab at University of Tubingen. Andreas Geiger was supported by the ERC Starting Grant LEGO-3D (850533) and the DFG EXC number 2064/1 - project number 390727645.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, pages 1209–1218, 2018. 7, 12, 16

[2] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8628–8638, 2021. 2, 3

[3] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 1

[4] 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 *CVPR*, 2016. 6, 7, 12

[5] Yutong Dai, Hao Lu, and Chunhua Shen. Learning affinity-aware upsampling for deep image matting. In *CVPR*, pages 6841–6850, 2021. 3

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 12

[7] Claude E Duchon. Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology (1962-1982)*, pages 1016–1022, 1979. 2

[8] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016. 3

[9] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 27, 2014. 12

[10] Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *CVPR*, pages 21795–21806, 2024. 1, 7, 12

[11] David F Fouhey, Wajahat Hussain, Abhinav Gupta, and Martial Hebert. Single image 3d without a single 3d image. In *ICCV*, pages 1053–1061, 2015. 12

[12] Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *ICLR*, 2024. 1, 2, 3, 4, 5, 6, 7, 12

[13] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T Freeman. Unsupervised semantic segmentation by distilling feature correspondences. *arXiv preprint arXiv:2203.08414*, 2022. 7, 12

[14] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, pages 991–998. IEEE, 2011. 13

[15] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *PAMI, volume=35, number=6, pages=1397–1409, year=2012, publisher=IEEE*. 3

[16] Haiwen Huang, Songyou Peng, Dan Zhang, and Andreas Geiger. Renovating names in open-vocabulary segmentation benchmarks. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 1, 6

[17] Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. *NeurIPS*, 33: 19545–19560, 2020. 6, 7, 12

[18] Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, Andre Araujo, Ricardo Martin-Brualla, Kaushal Patel, Daniel Vlasic, Vittorio Ferrari, Ameesh Makadia, Ce Liu, Yuanzhen Li, and Howard Zhou. NAVI: Category-agnostic image collections with high-quality 3d shape and pose annotations. In *NeurIPS*, 2023. 7, 12, 16

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 13

[20] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, pages 9799–9808, 2020. 3

[21] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, pages 4015–4026, 2023. 1, 2, 4, 5, 6, 12, 13

[22] Johannes Kopf, Michael F Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. *ACM TOG*, 26(3): 96–es, 2007. 3, 5

[23] Mengcheng Lan, Chaofeng Chen, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Proxyclip: Proxy attention improves clip for open-vocabulary segmentation. In *ECCV*, 2024. 6, 7, 12

[24] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European conference on computer vision*, pages 280–296. Springer, 2022. 1

[25] Kai-En Lin, Yen-Chen Lin, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. Vision transformer for nerf-based view synthesis from a single input image. pages 806–815, 2023. 3

[26] 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 *ECCV*, pages 740–755. Springer, 2014. 6, 12

[27] Qin Liu, Zhenlin Xu, Gedas Bertasius, and Marc Niethammer. Simpleclick: Interactive image segmentation with simple vision transformers. In *ICCV*, pages 22290–22300, 2023. 1, 6, 7, 12, 13

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 12

[29] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Index networks. *IEEE TPAMI*, 44(1):242–255, 2020. 3

[30] Hao Lu, Wenze Liu, Hongtao Fu, and Zhiguo Cao. Fade: Fusing the assets of decoder and encoder for task-agnostic upsampling. In *ECCV*, pages 231–247. Springer, 2022. 3

[31] Hao Lu, Wenze Liu, Zixuan Ye, Hongtao Fu, Yuliang Liu, and Zhiguo Cao. Sapa: Similarity-aware point affiliation for feature upsampling. *NeurIPS*, 35:20889–20901, 2022. 3

[32] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, pages 416–423. IEEE, 2001. 6, 13

[33] Sky McKinley and Megan Levine. Cubic spline interpolation. *College of the Redwoods*, 45(1):1049–1060, 1998. 2

[34] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3, 13

[35] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 3

[36] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10):e3, 2016. 3

[37] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. 2024. Featured Certification. 1, 2, 3, 6, 12, 15

[38] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, pages 724–732, 2016. 6, 13

[39] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. 2, 7, 12, 17

[40] Gilles Puy, Spyros Gidaris, Alexandre Boulch, Oriane Siméoni, Corentin Sautier, Patrick Pérez, Andrei Bursuc, and Renaud Marlet. Three pillars improving vision foundation model distillation for lidar. In *CVPR*, pages 21519–21529, 2024. 5

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. pages 8748–8763. PmLR, 2021. 1, 3, 6, 12, 13

[42] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, pages 12179–12188, 2021. 7

[43] Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Am-radio: Agglomerative vision foundation model reduce all domains into one. In *CVPR*, pages 12490–12500, 2024. 1, 6, 13

[44] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *CVPR*, pages 13009–13018, 2024. 4

[45] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1, 2

[46] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024. 4

[47] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 3

[48] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. " grabcut" interactive foreground extraction using iterated graph cuts. *ACM TOG*, 23(3):309–314, 2004. 6, 13

[49] Wenzhe Shi, Jose Caballero, Lucas Theis, Ferenc Huszar, Andrew Aitken, Christian Ledig, and Zehan Wang. Is the deconvolution layer the same as a convolutional layer? *arXiv preprint arXiv:1609.07009*, 2016. 3

[50] Konstantin Sofiiuk, Olga Barinova, and Anton Konushin. Adaptis: Adaptive instance selection network. In *ICCV*, pages 7355–7363, 2019. 13

[51] Konstantin Sofiiuk, Ilya A Petrov, and Anton Konushin. Reviving iterative training with mask guidance for interactive segmentation. In *ICIP*, pages 3141–3145. IEEE, 2022. 13

[52] Saksham Suri, Matthew Walmer, Kamal Gupta, and Abhinav Shrivastava. Lift: A surprisingly simple lightweight feature transform for dense vit descriptors. In *ECCV*, pages 110–128. Springer, 2024. 1, 2, 3, 4, 6, 7, 12

[53] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *CVPR*, pages 10208–10217, 2024. 3

[54] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. 1

[55] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil

Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. 1

[56] Narek Tumanyan, Assaf Singer, Shai Bagon, and Tali Dekel. Dino-tracker: Taming dino for self-supervised point tracking in a single video. In *ECCV*, pages 367–385. Springer, 2024. 1

[57] Jiaqi Wang, Kai Chen, Rui Xu, Ziwei Liu, Chen Change Loy, and Dahua Lin. Carafe: Content-aware reassembly of features. In *ICCV*, pages 3007–3016, 2019. 3

[58] Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *CVPR*, pages 14543–14553, 2022. 5

[59] Monika Wysoczańska, Oriane Siméoni, Michaël Ramamonjisoa, Andrei Bursuc, Tomasz Trzciński, and Patrick Pérez. Clip-dinoiser: Teaching clip a few dino tricks for open-vocabulary semantic segmentation. In *ECCV*, pages 320–337. Springer, 2024. 5

[60] Jianglong Ye, Naiyan Wang, and Xiaolong Wang. Featurenerf: Learning generalizable nerfs by distilling foundation models. In *ICCV*, pages 8962–8973, 2023. 3, 13

[61] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. 2021 ieee. In *CVPR*, 2020. 3, 13

[62] Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. *Advances in Neural Information Processing Systems*, 36:32215–32234, 2023. 1

[63] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, pages 11975–11986, 2023. 1

[64] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, pages 633–641, 2017. 6, 12