



<https://wimmerth.github.io/anyup>

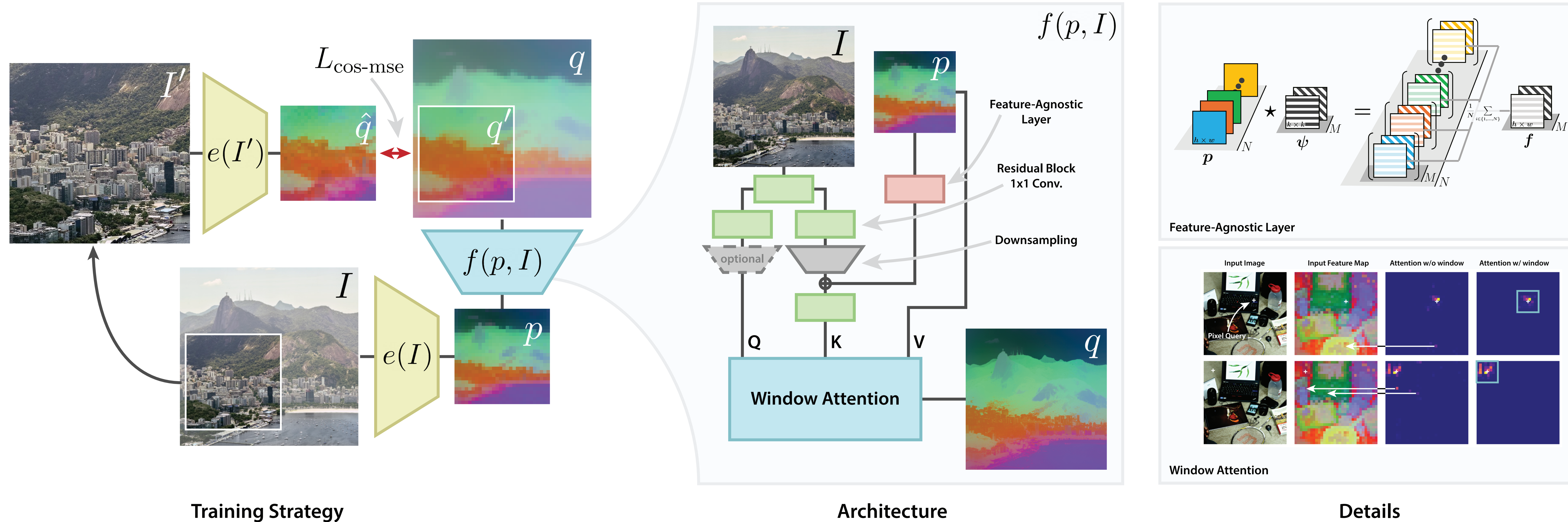
Upsample Any Feature at Any Resolution for Any Task without Retraining!

AnyUp: Universal Feature Upsampling

Thomas Wimmer^{1,2}, Prune Truong³, Marie-Julie Rakotosaona³, Michael Oechsle³, Federico Tombari^{3,4}, Bernt Schiele¹, Jan Eric Lenssen¹

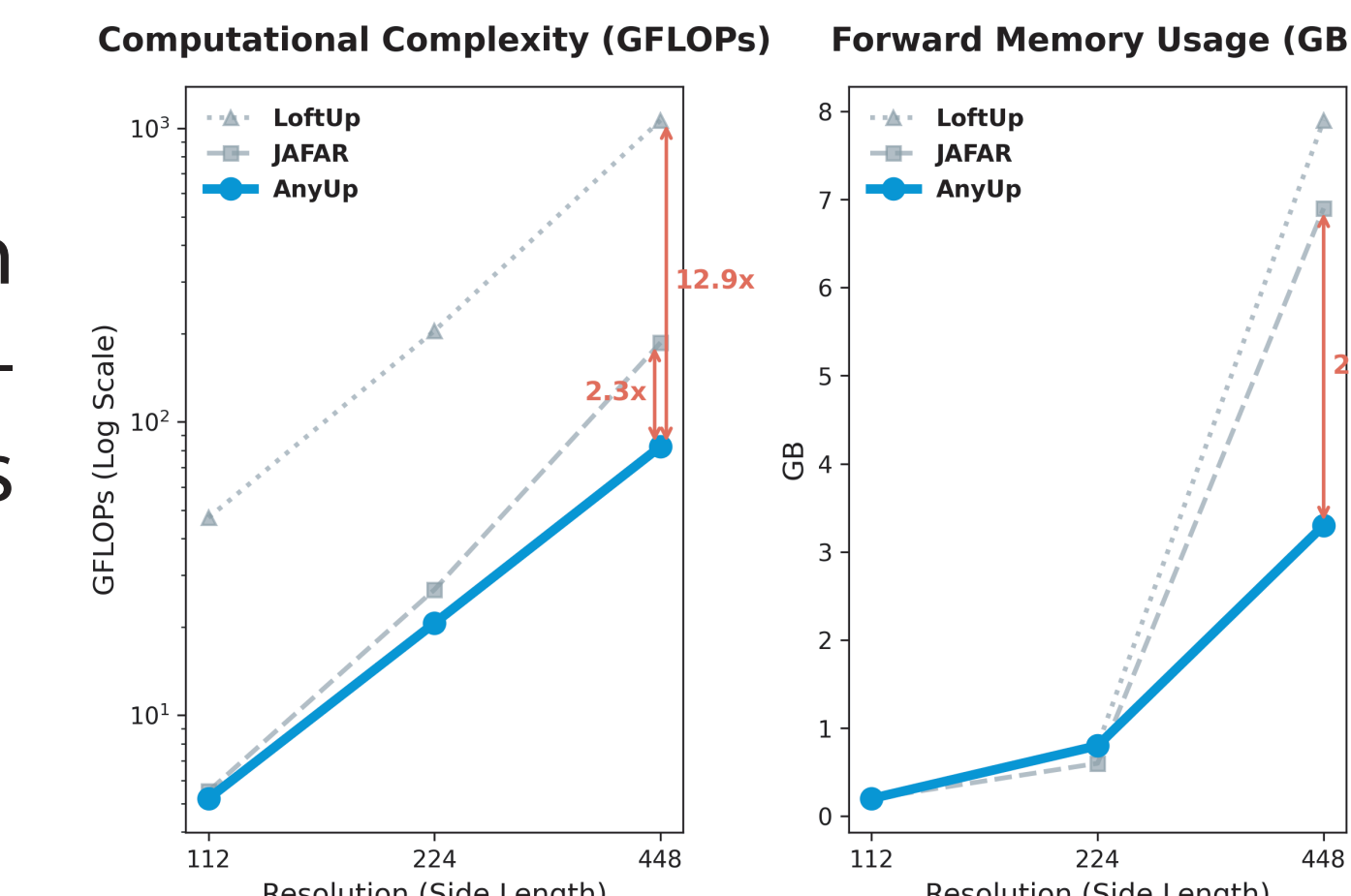
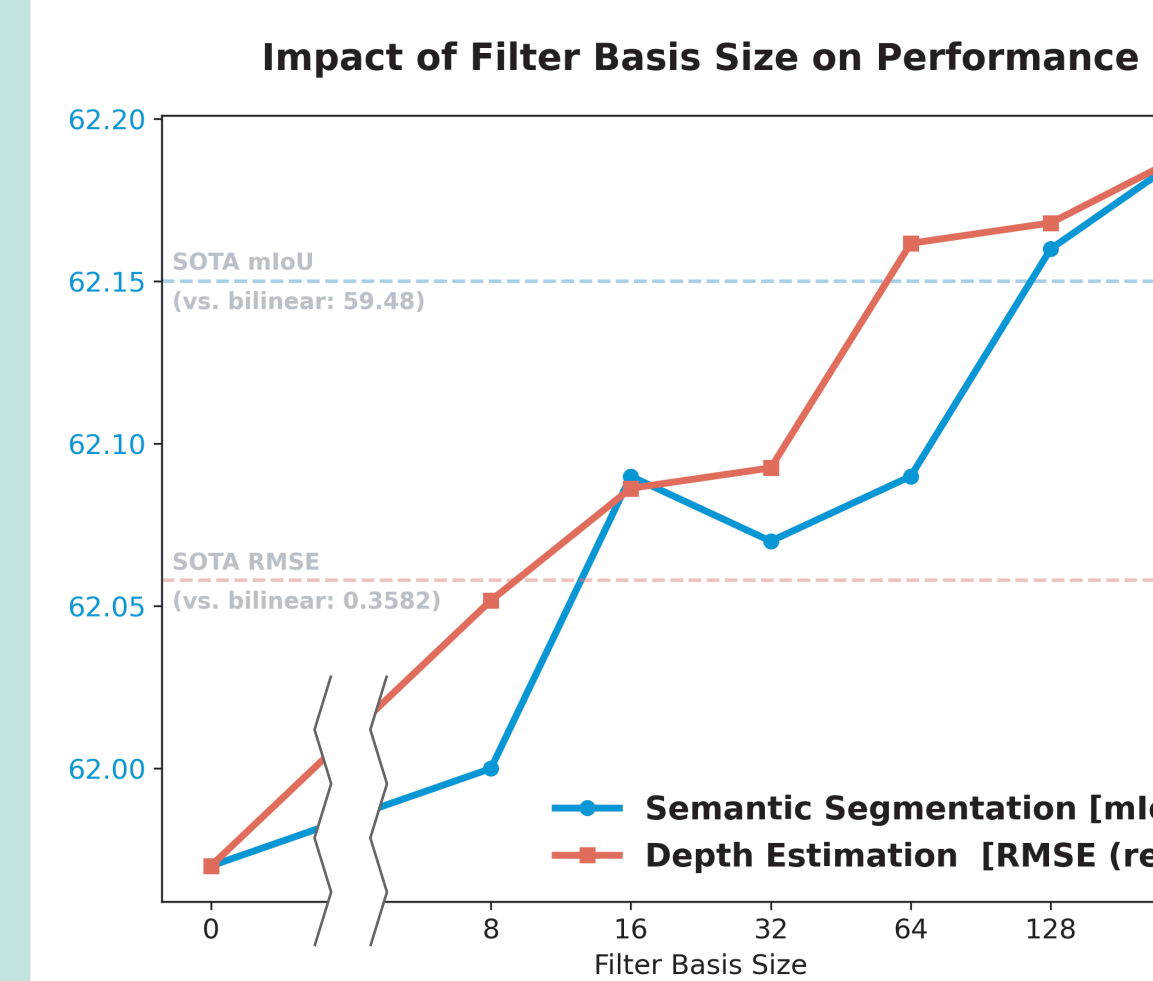
¹Max Planck Institute for Informatics, ²ETH Zurich, ³Google, ⁴Technical University of Munich

Method



Analysis

AnyUp is more efficient than previous attention-based feature upsamplers through its usage of window attention.

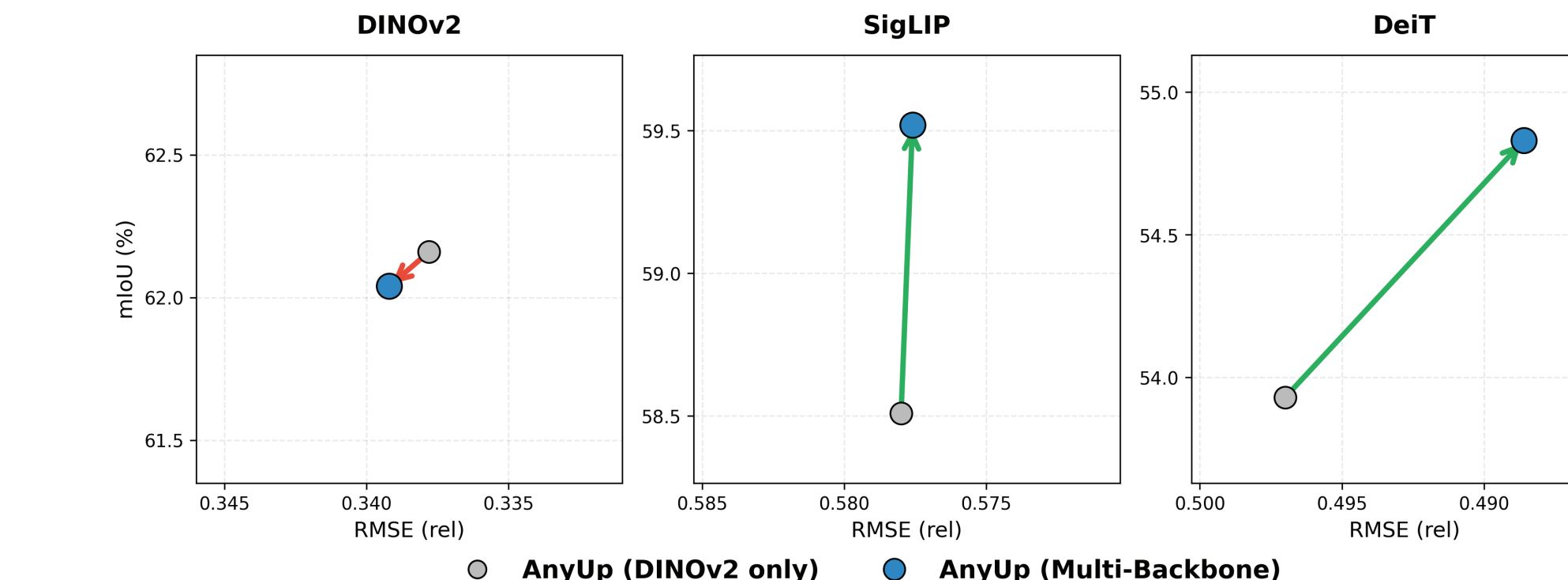


“Only looking at the RGB image” by matching colors and positions between a high-resolution image and its downsampled version is a surprisingly strong baseline.

AnyUp retains the input feature space while improving up-sampling quality.

	Semantic Segmentation mIoU (↑)	Acc. (↑)	Depth Estimation RMSE (↓)	δ_1 (↑)
Bilinear	29.73	73.32	0.506	0.816
Guided Filter	37.54	72.25	0.518	0.813
FeatUp	40.19	74.05	0.504	0.818
LoftUp	4.27	46.58	0.765	0.802
JAFAR	39.06	73.73	0.503	0.815
AnyUp	40.83	74.94	0.498	0.822

Training on multiple feature extractors improves generalization while achieving comparable performance on seen backbones.



Limitations:

- Upsampled features are a linear combination of input features
→ Sub-patch-level information in the features is not leveraged
- Feature upsampling is post-hoc and bound to the quality of extracted low-resolution features.

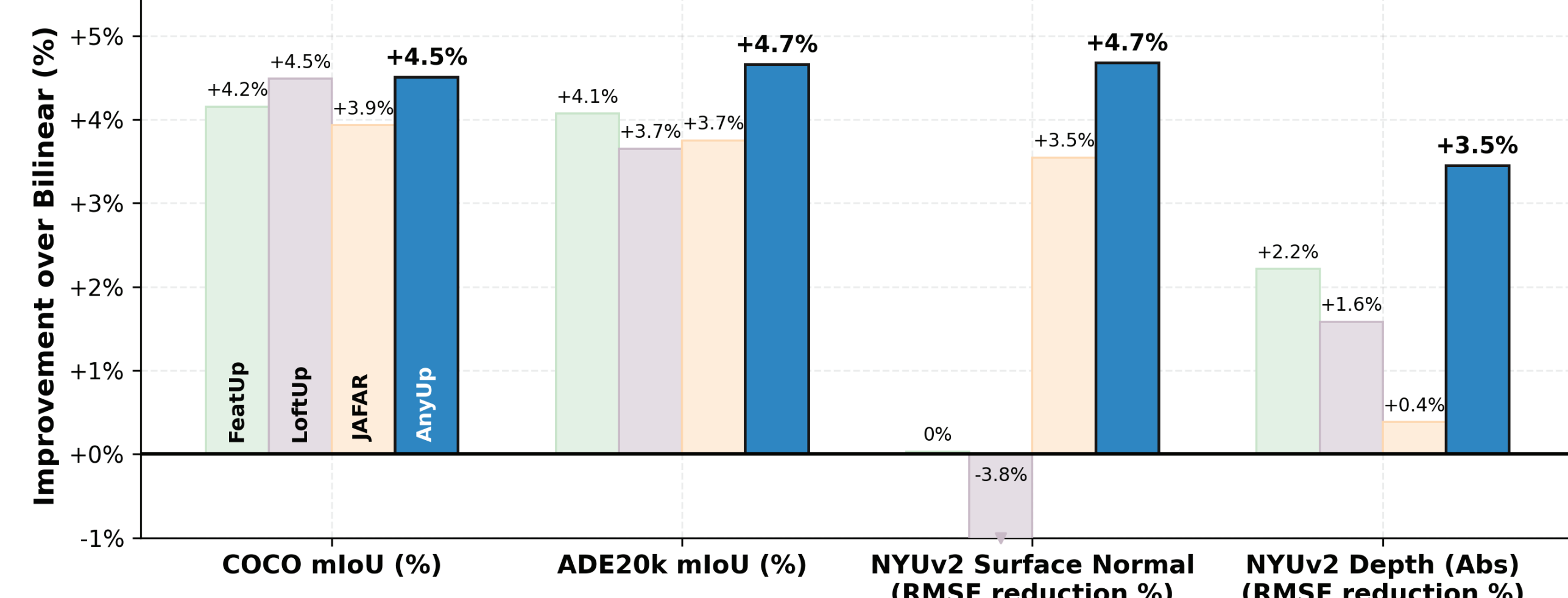
Usage:

```
# load the AnyUp upsampler model
upsampler = torch.hub.load('wimmerth/anyup', 'anyup')

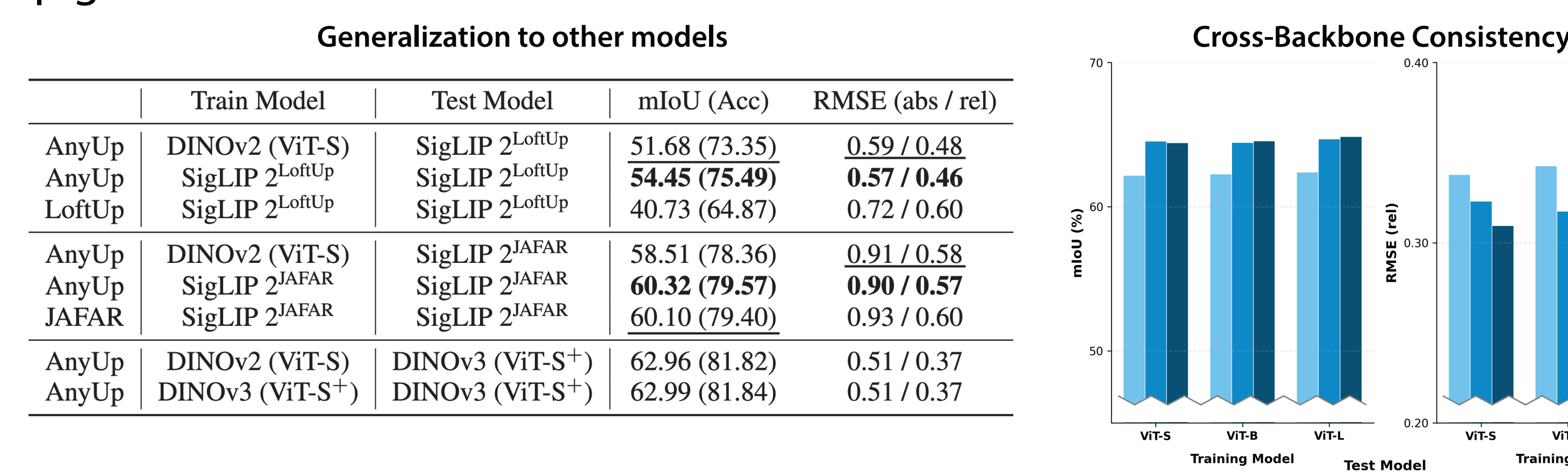
# upsampled high-resolution features (B, C, H, W)
hr_features = upsampler(hr_image, lr_features)
```

Results

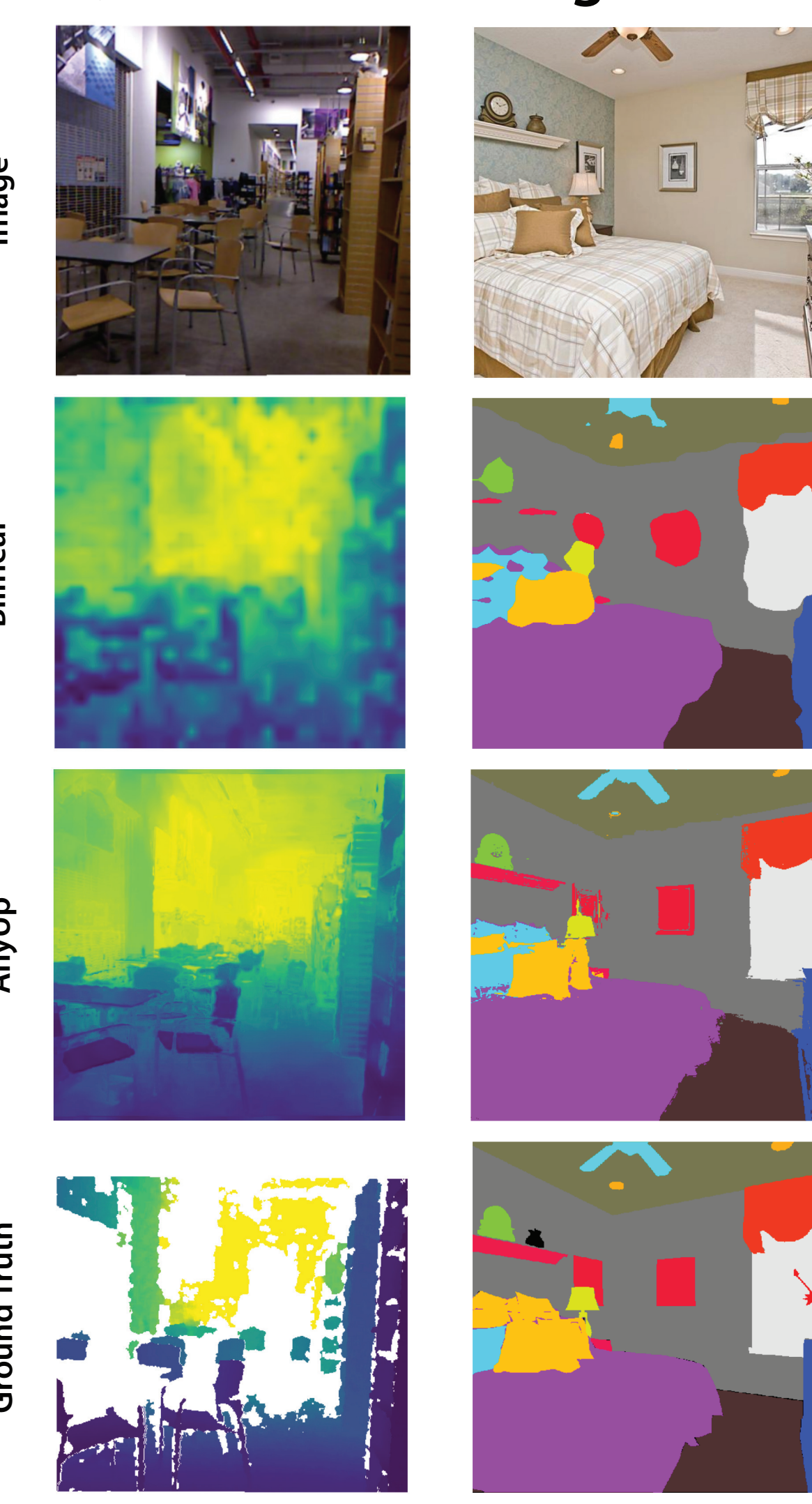
AnyUp beats feature-specific upsamplers in downstream linear probing.



AnyUp generalizes well to unseen feature extractors at test time.



Qualitative Probing Results



RGB Image



Low Resolution

AnyUp