

Scene Understanding for Autonomous Drone Delivery (SUADD)

Team seg-dep

Mykola Lavreniuk, Nivedita Rufus,
Unnikrishnan R Nair

Scene Understanding for Autonomous Drone



Semantic Segmentation & Depth Estimation

🏆 \$50,000 Cash
Prize Pool

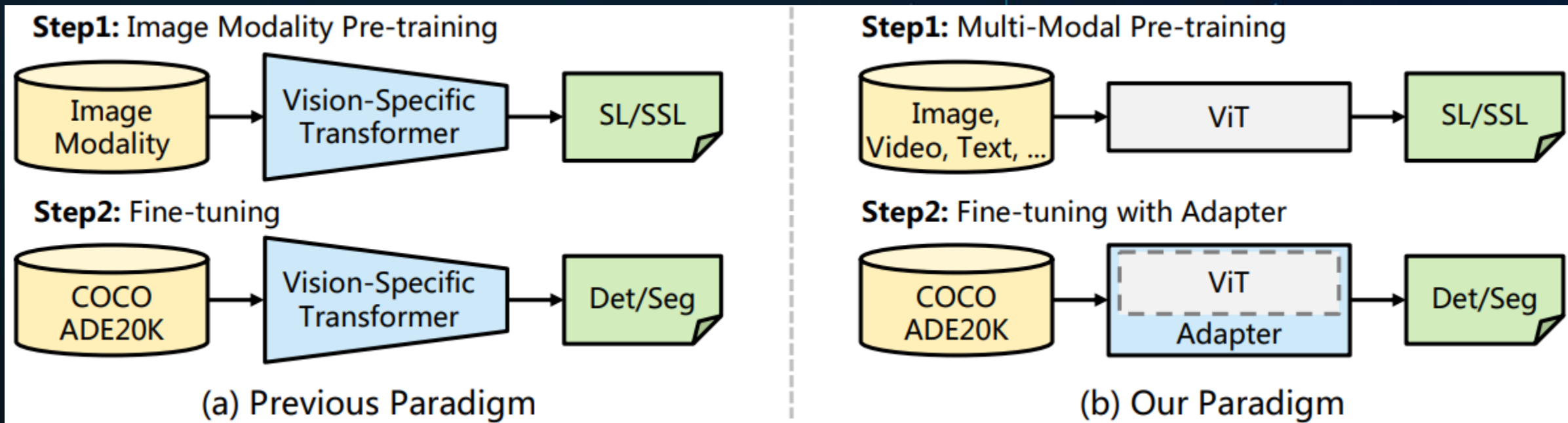


Semantic Segmentation Challenge

3rd place solution

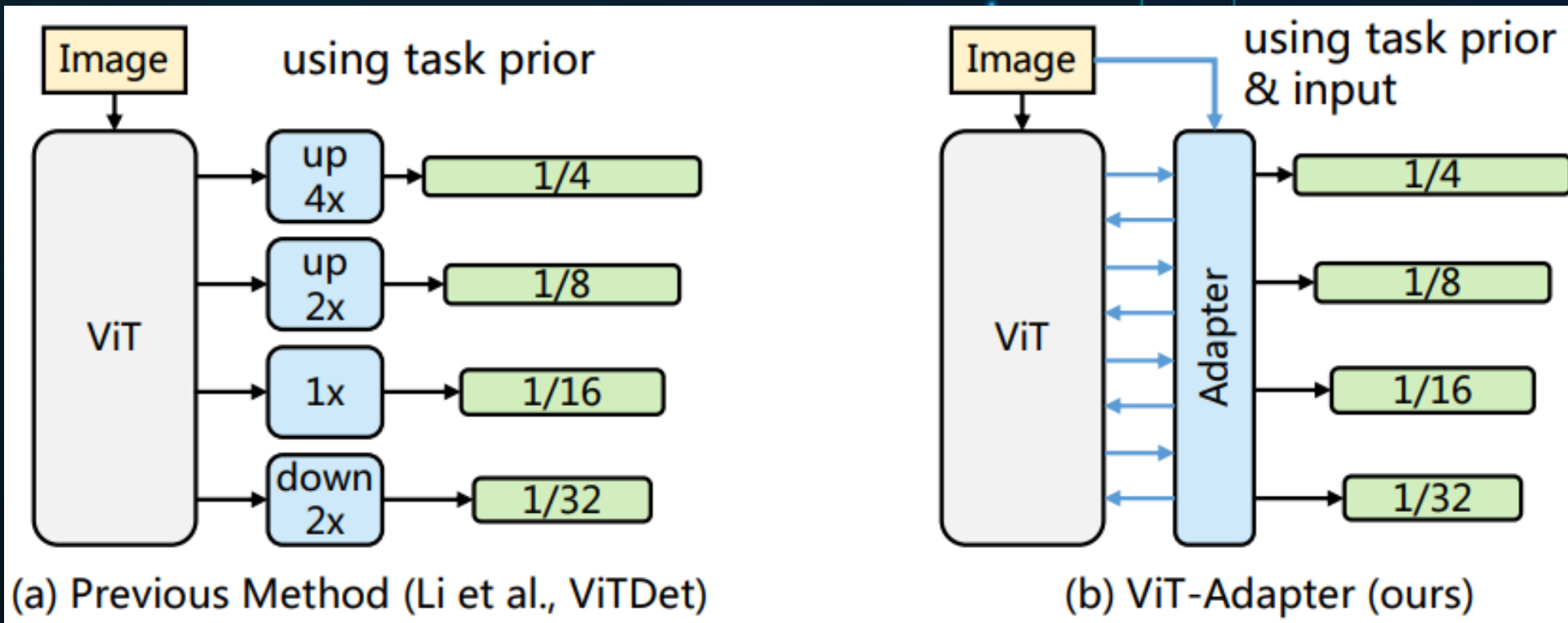
ViT-Adapter

Designed a spatial prior module and two feature interaction operations, to inject the image prior without redesigning the architecture of ViT. They can supplement the missing local information and reorganize fine-grained multi-scale features for dense prediction tasks

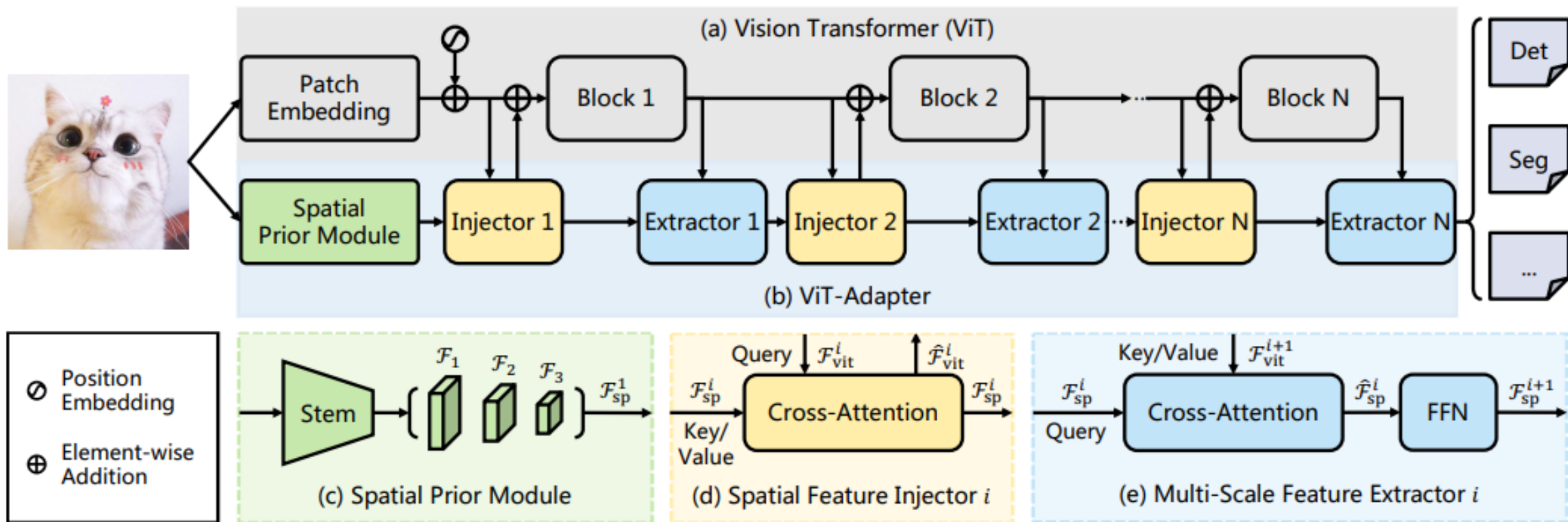


ViT-Adapter

plain backbone, w/ simple feature pyramid



ViT-Adapter



Solution summary

Backbone: ViT-Adapter-L

Method: Mask2Former

Pretrain: BEiTv2-L+COCO and after on ADE20K

Finetune: SUADD train + val

Train images: 896x896 randomcrop from (1550x2200)

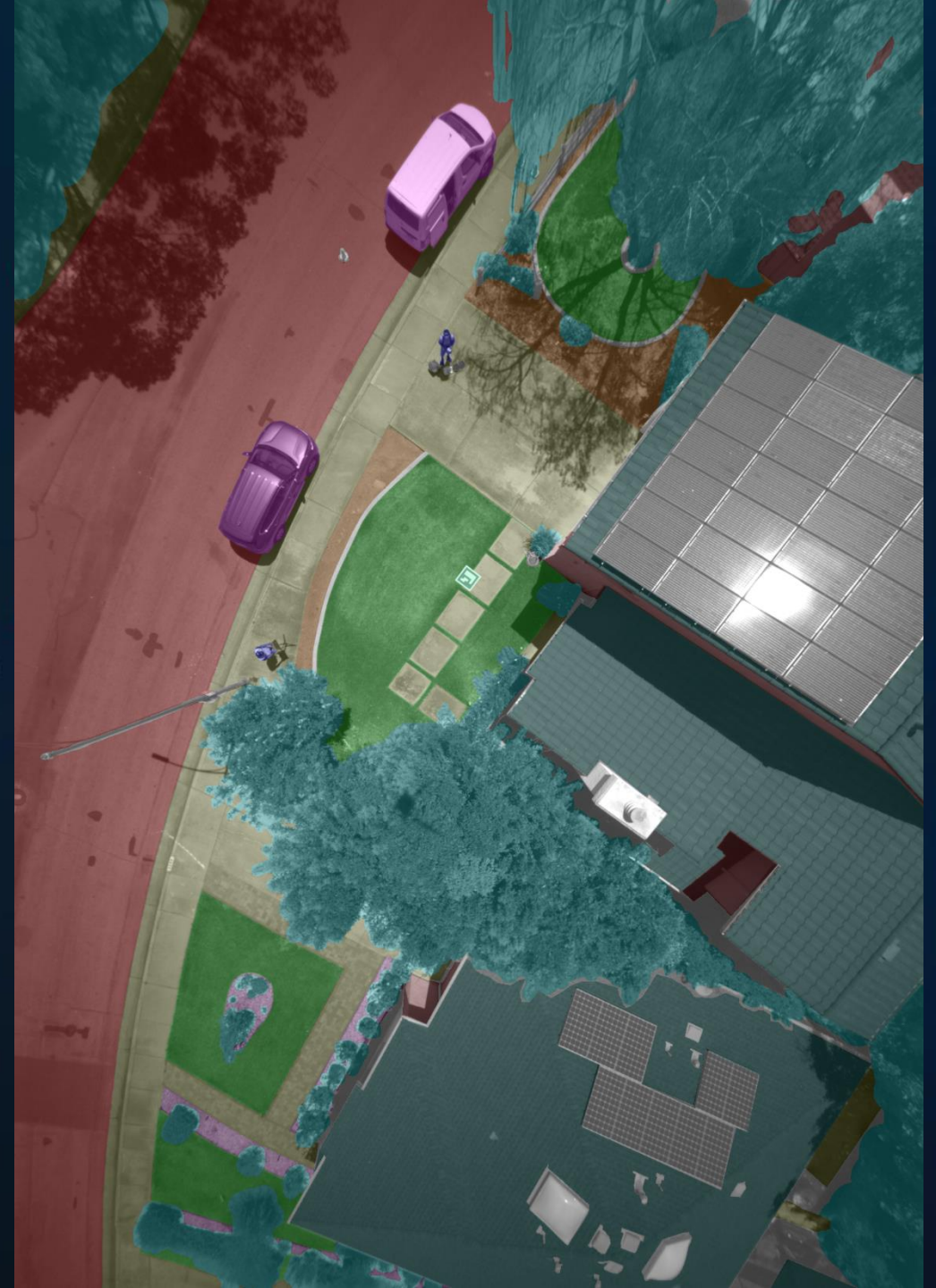
Test images: 896x896 crop with same stride from (896x1792)

Augmentations: standard from ViT-Adapter-L

Optimizer: AdamW, lr = $2e-05$, iterations = 20k

Other tricks:

- ✓ EMA,
- ✓ Model soup,
- ✓ RGB creation on train based on copy same image, on inference - color augmentations.



What **did not** work for us

Models and methods: EVA, InternImage, SegFormer

Pretrain: Mapillary, Cityscapes

Train and test images: different other settings

Augmentations: different other settings

Other tricks:

- Larger weights for rare and low accurate classes
- Use only train dataset for training
- Split train and validation in other way



Scene Understanding for Autonomous Drone



Semantic Segmentation & Depth Estimation

🏆 \$50,000 Cash
Prize Pool



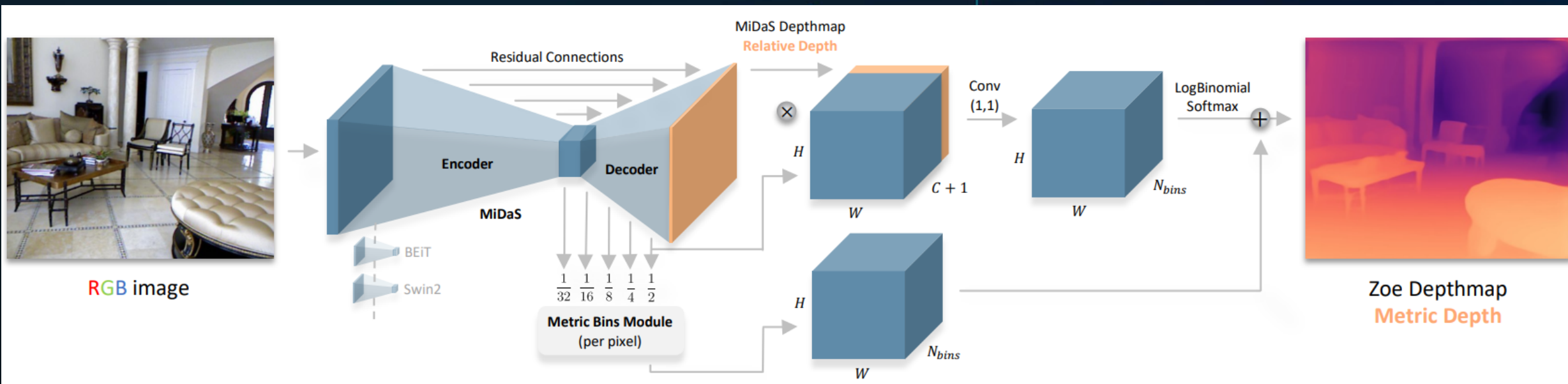
Mono-depth Estimation

2rd place solution

ZoeDepth

Zero-shot Transfer by Combining Relative and Metric Depth

ZoeDepth, bridges the gap between relative and metric depth estimation. In the first stage, it pre-trains an encoder-decoder using relative depth datasets. In the second stage, domain-specific heads added based on new metric bins module to the decoder and fine-tune the model on datasets for metric depth prediction.



Solution summary

Backbone: BEiT384-L

Model: ZoeD-M12-N

Pretrain: relative depth on M12 dataset and after on NYU Depth v2

Finetune: SUADD train + val

Train images: 768x512 and 896x592

Test images: 768x512 and 896x592

Augmentations: standard from Zoe + rotate + translate + both flips

Optimizer: AdamW, lr = 0.000018, epochs = 20

Other tricks:

- ✓ Ensemble of different resolutions + TTA (vertical flip),
- ✓ Remove image padding for inference,
- ✓ Model soup,
- ✓ RGB creation on train based on copy same image, on inference - color augmentations.

What **did not** work for us

Models and methods: ZoeD-M12-NK, MIDAS

Pretrain: on both NYU Depth v2 and KITTI

Train and test images: larger image resolution

Augmentations: random crop, CutFlip (URCDC-Depth paper)

Other tricks:

- Edge enhancement,
- Horizontal flip in TTA,
- Use only train dataset for training,
- Split train and validation in other way.



* We have not be able to run the SOTA model - **VPD**
(Visual Perception with a pre-trained Diffusion model)

Thank you!



Final position on the leaderboard.
[Link to code](#)




bsn 
2.0



seg-dep 
5.0



gs-sai 
9.0

