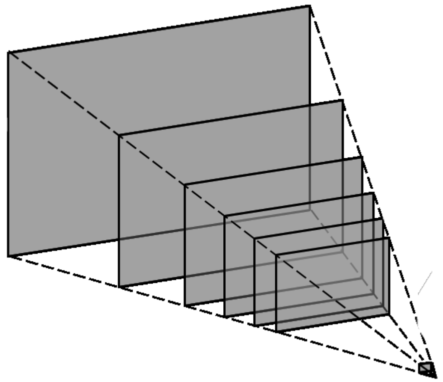
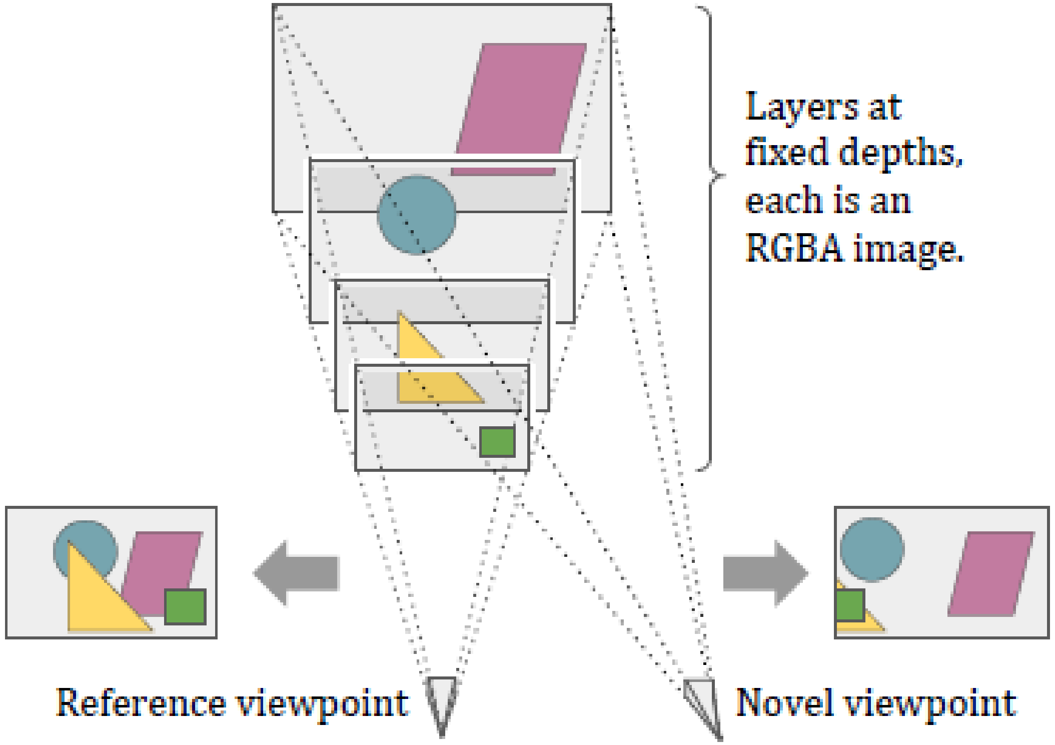
Occupancy as condition input

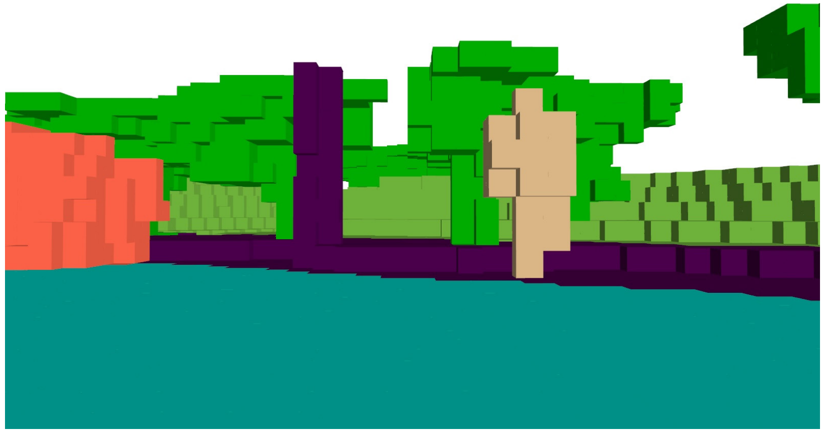
- Our solution: use multi plane images (MPIs) to enforce spatial alignment
- Multi plane images are layers of images in different depth.
- Spatial alignment: the projection of MPIs are real images



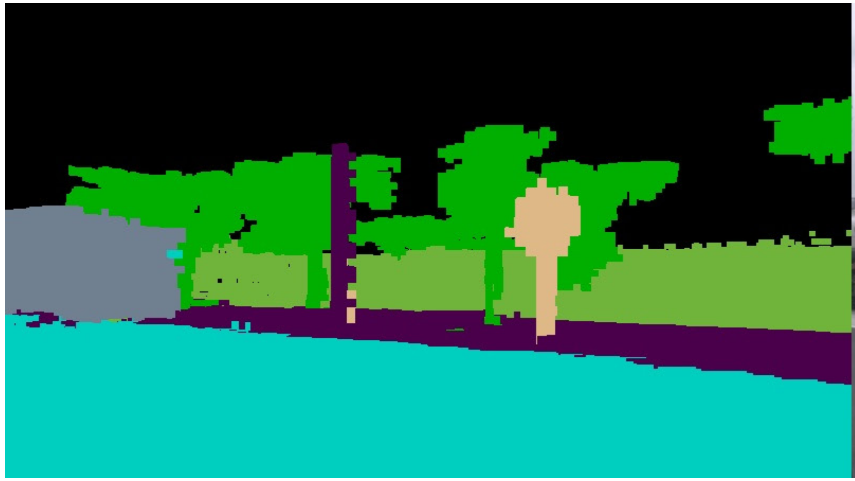


Occupancy as condition input

- 1. map sparse occupancy to dense voxel
- Point cloud to voxel



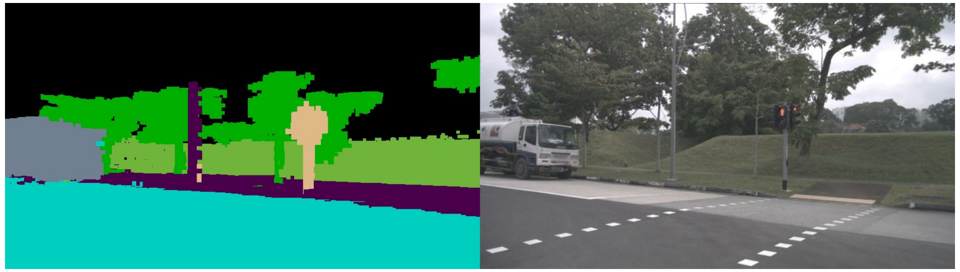
Occupancy rendering



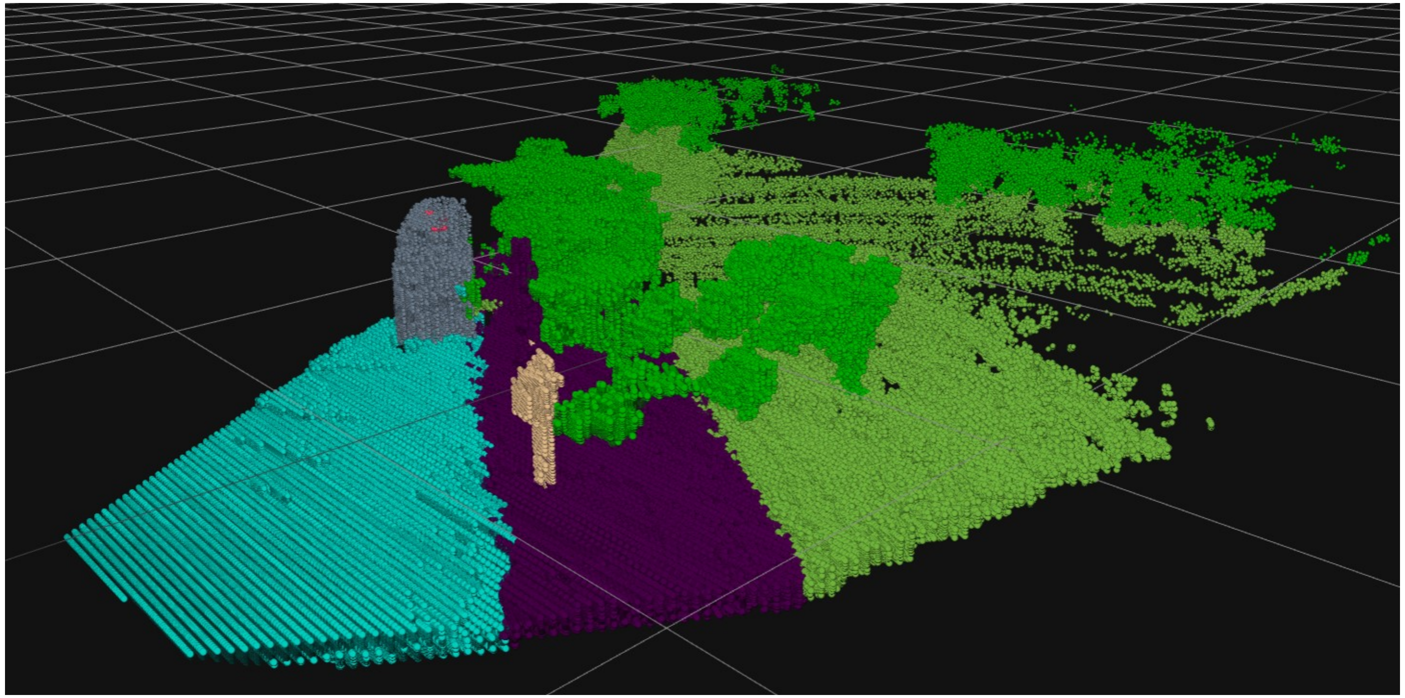
Occupancy projection

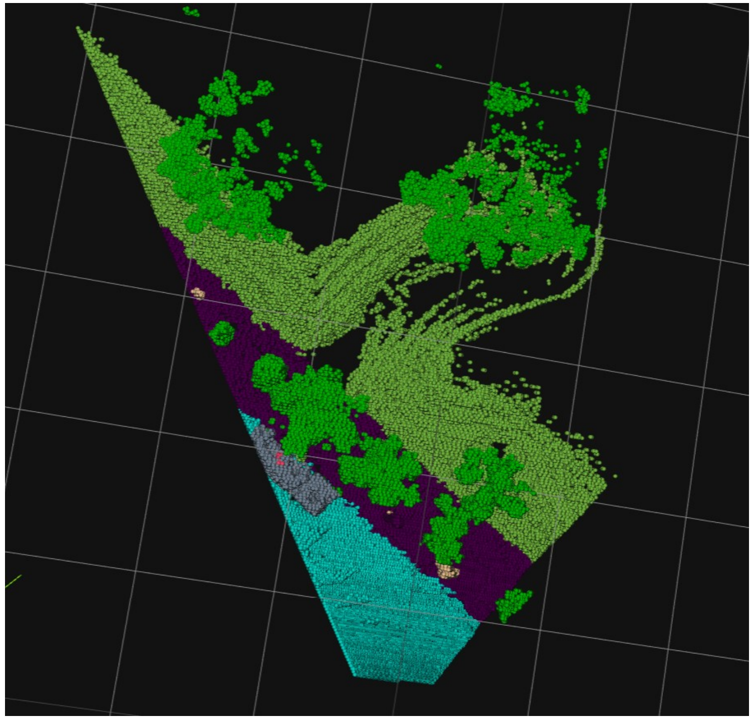
Real image

Occupancy as condition input



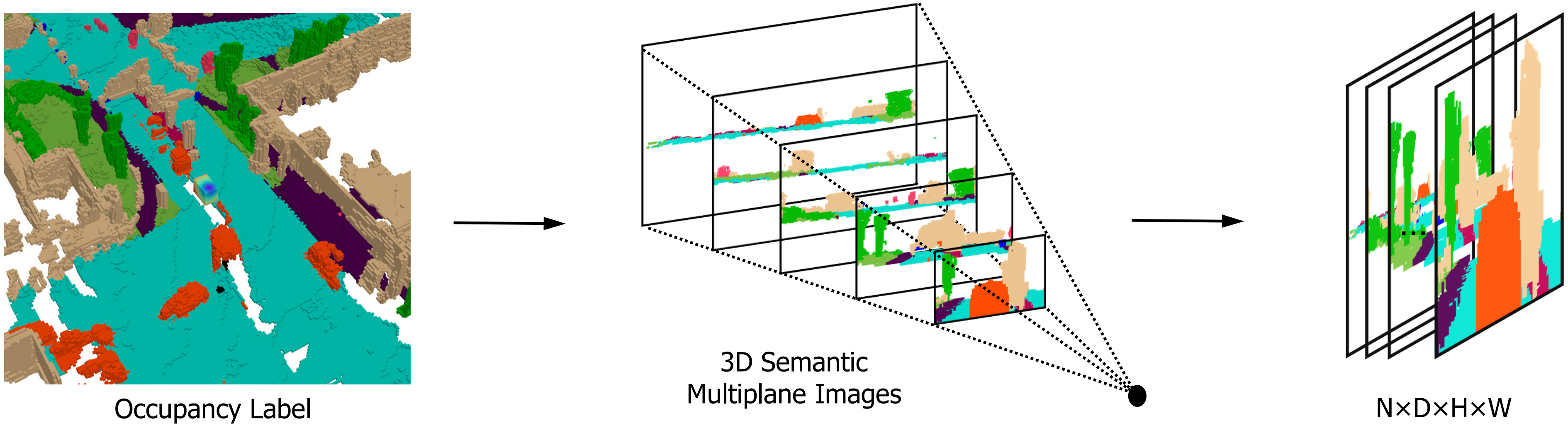
- 1. map sparse occupancy to dense voxel
- 2. cast camera rays to form frustum (multi-channel images)
- 3. interpolation, output is H*W*D*1, D is the plane number, 1 is semantic label





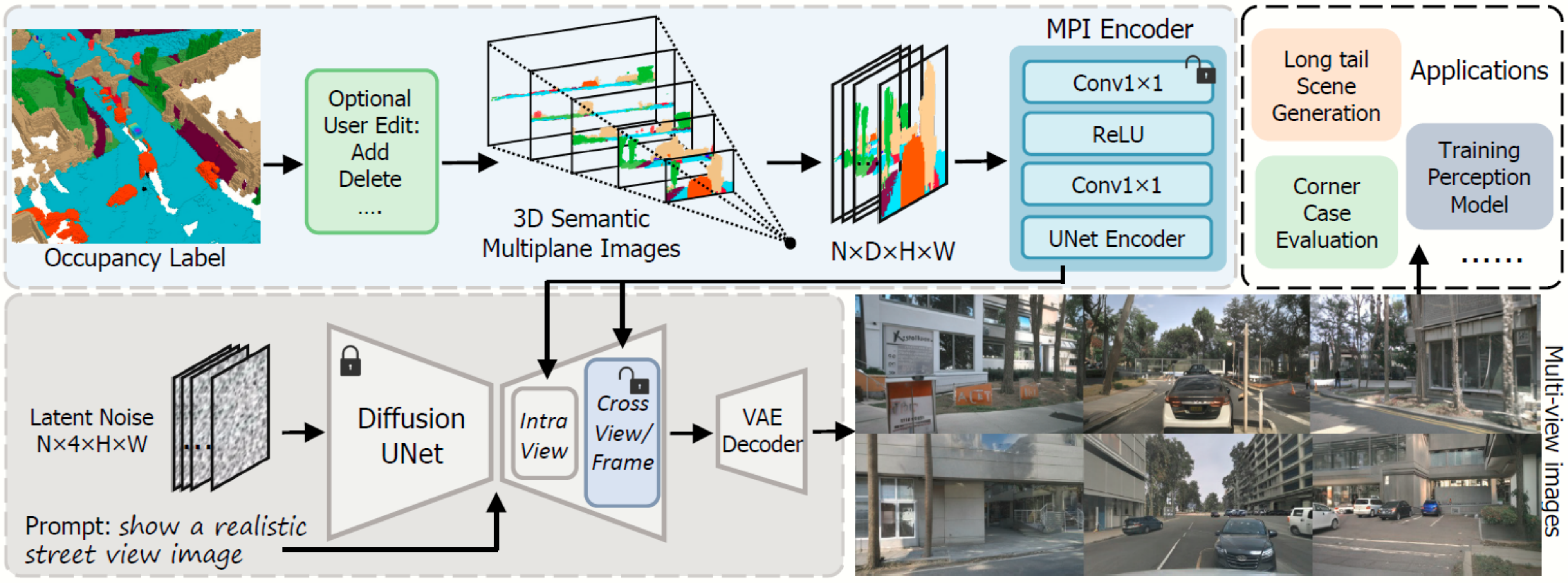
Summary

- We cast 256 planes from 0m to 50m
- For example, the first plane contain semantics from 0m to 50/256m
- We set the D dimension analogous to latent dimension



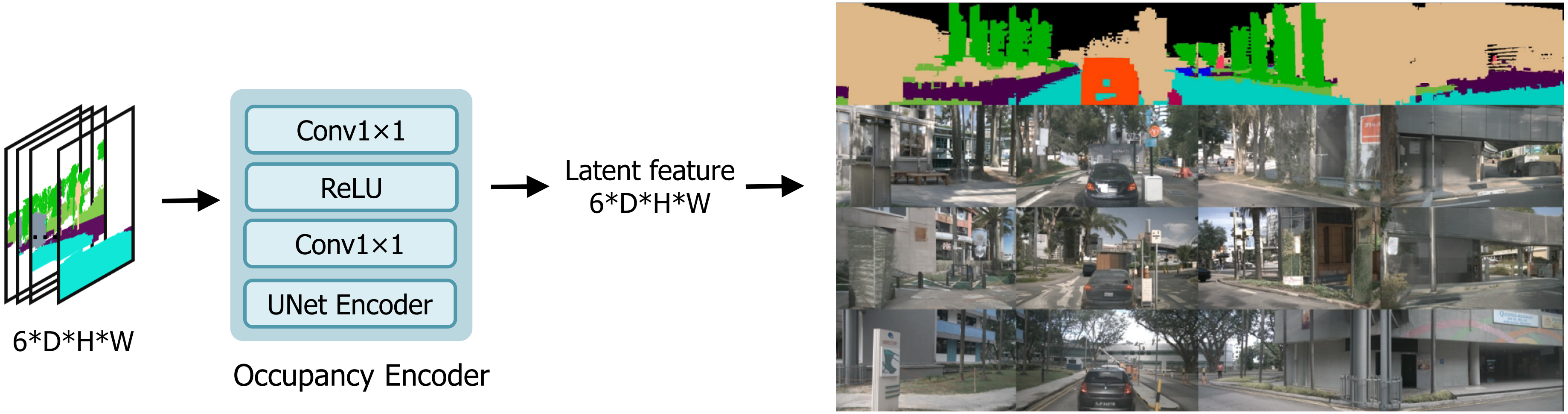
Limitation of prior work

- Fix vocabulary generation to open vocabulary generation
- Input of SyntheOcc: class label (0,1,2,3, …)
- Control image: N*D*H*W



Design of occupancy encoder

- The MPIs is in camera frustum space, not Euclid space. Applying convolution with kernel size > 1 will break geometric relation. (axis-align issue)
- Solution: apply 1x1 conv, keep the condition size to latent dimension, provide spatial alignment



6*D*H*W

Conv1×1

ReLU

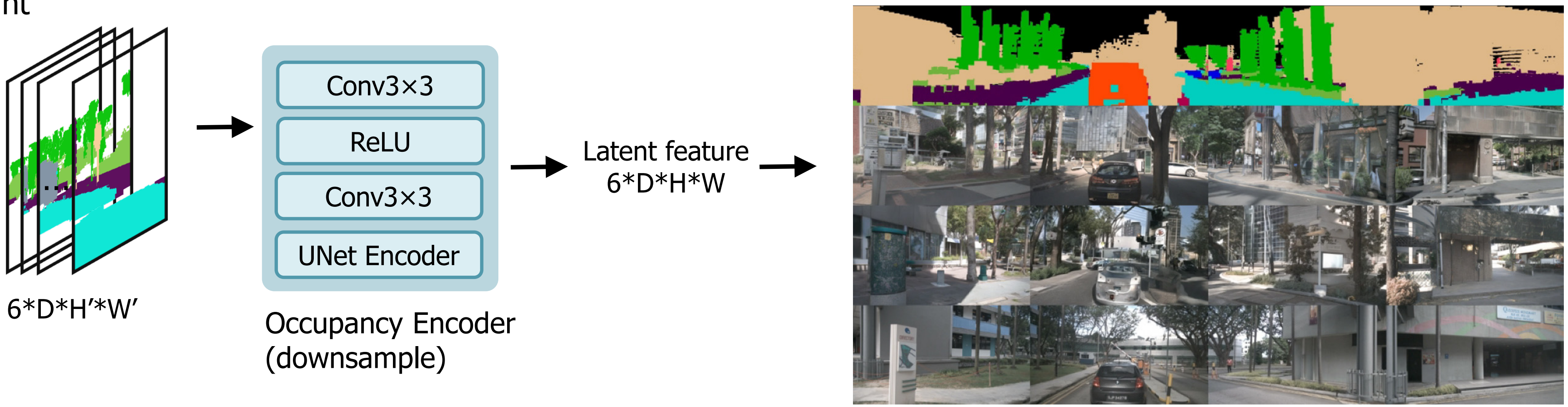
Conv1×1

UNet Encoder

Occupancy Encoder

Latent feature
6*D*H*W

H = H’ // 8
H is height in vae space
H’ is original image height



6*D*H’*W’

Conv3×3

ReLU

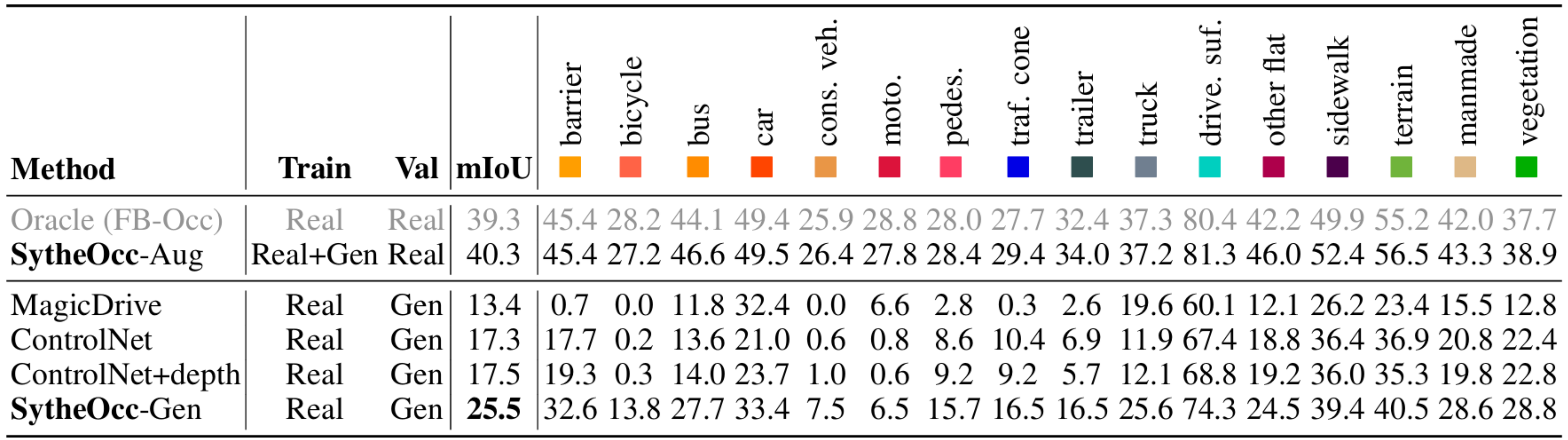
Conv3×3

UNet Encoder

Occupancy Encoder
(downsample)

Latent feature
6*D*H*W

Results



Method	Train	Val	mIoU	barrier 🟧	bicycle 🟥	bus 🟧	car 🟥	cons. veh. 🟧	moto. 🟥	pedes. 🟥	traf. cone 🟦	trailer ⬛	truck ⬜	drive. suf. 🟦	other flat 🟥	sidewalk 🟪	terrain 🟩	manmade 🟫	vegetation 🟩
Oracle (FB-Occ)	Real	Real	39.3	45.4	28.2	44.1	49.4	25.9	28.8	28.0	27.7	32.4	37.3	80.4	42.2	49.9	55.2	42.0	37.7
SytheOcc-Aug	Real+Gen	Real	40.3	45.4	27.2	46.6	49.5	26.4	27.8	28.4	29.4	34.0	37.2	81.3	46.0	52.4	56.5	43.3	38.9
MagicDrive	Real	Gen	13.4	0.7	0.0	11.8	32.4	0.0	6.6	2.8	0.3	2.6	19.6	60.1	12.1	26.2	23.4	15.5	12.8
ControlNet	Real	Gen	17.3	17.7	0.2	13.6	21.0	0.6	0.8	8.6	10.4	6.9	11.9	67.4	18.8	36.4	36.9	20.8	22.4
ControlNet+depth	Real	Gen	17.5	19.3	0.3	14.0	23.7	1.0	0.6	9.2	9.2	5.7	12.1	68.8	19.2	36.0	35.3	19.8	22.8
SytheOcc-Gen	Real	Gen	**25.5**	32.6	13.8	27.7	33.4	7.5	6.5	15.7	16.5	16.5	25.6	74.3	24.5	39.4	40.5	28.6	28.8

Table 1: Downstream evaluation on the **nuScenes-Occupancy** validation set. Based on the used train and val data, two types of settings are reported. The first is to use generated training set to augment the real training set, and evaluate on the real validation set, denoted as Aug. The second is to use pretrained models trained on the real training datasets to test on the generated validation set, denoted as Gen.