HEAL-SWIN: A Vision Transformer On The Sphere

Supplementary Material

A. Datasets

WoodScape and SynWoodScape For our experiments we use the WoodScape [49] and the SynWoodScape [45] dataset. Note that we used the 2k samples which were published at the time of writing[5], instead of the full 80k samples. In all experiments, we split the available samples randomly (but consistently across models and runs) into 80% training data and 20% validation data. For the semantic segmentation task, we use the 10 classes for which semantic masks are provided in the WoodScape dataset and two different subsets of the 25 classes for the SynWoodScape dataset. Table 4 shows the relation between the 25 original classes and the classes in our two subsets. See Figure 9 for the class prevalences in the different datasets. Examples of the inconsistent semantic masks in the WoodScape dataset discussed in the main text can be found in Figure 10.

In the 2021 CVPR competition for segmentation of the WoodScape dataset [43], pixels that are labeled with the dominant *void* class in the ground truth were excluded from the mIoU used for ranking. Therefore, many teams excluded the void class from their training loss, resulting in random predictions for large parts of the image. This shortcoming was noted in [43], but the evaluation score could not be changed after the competition had been published. Given these circumstances, we decided to include the *void* class into our training loss but exclude it from the mean over classes in the mIoU to more accurately reflect the performance of our models on the more difficult classes. However, this also means that our results cannot be directly compared to the results of the competition.

The same problem does not arise for the SynWoodScape dataset and our two variants since their class lists include all major structures in the image, leading to a much reduced prevalence of the void class. Therefore, we include all classes in the mIoU for these datasets.

Stanford 2D-3D-S The Stanford 2D-3D-Semantic dataset[6] [1] consists of 1413 omni-directional RGB-D images of indoor scenes from six different buildings. These six areas are used to create an official three fold cross validation split. Each area has complete semantic segmentation annotations for 13 object classes and 2 "void" classes: "background" and "unknown". For all tasks we map all regions of the "unknown" class to the "background" class. We normalize all RGB-D input channels individually and ignore the background class during training and evaluation, in line with [29].

In contrast to the depth values and the semantic segmentation ground truth, which exist for the entire sphere, the image RGB values are non-zero only between -60° and 60°, see Figure 11 for a visualization. For training we chose to map the polar regions where no non-zero RGB data is present to background which is ignored during training. In that way the only areas the network is trained on are those where both RGB and depth values are present.

Removing the polar ground truth affects the computation of the IoU; specifically for classes that are heavily correlated with these areas. In the case of the Stanford 2D-3D-S dataset both the ceiling and floor classes often have large overlap with the polar regions and hence those classes have a disproportionately large union, which, in turn, will reduce their IoU-score.

For completeness, we report the IoU for the same trained instance of the mode on both cases: when the polar regions are mapped to background and when the segmantic ground truth is kept. The per class IoU metrics for both cases are presented in Table 5.

In addition, there are degenerate samples in the Stanford 2D-3D-S dataset in which the ground truth consists only of the background class. One such sample is shown in Figure 14.

B. Additional experiment: Spherical image classification

A common low-resolution dataset on which performance of spherical models is measured is a spherical projection of MNIST. We project the MNIST digits onto a HEALPix grid of $n_{\text{side}} = 16$, corresponding to 3072 input pixels, less than the 3600 input pixels often used on the Driscoll-Healy grid. On this dataset, we train a HEAL-SWIN classifier consisting of 10 transformer layers followed by three fully-connected layers resulting in a model with about 62k parameters.

We train and evaluate our model both on unrotated data (NR/NR modality) and on rotated data (R/R modality). In the NR/NR modality, the task is very simple and most spherical models (including ours) reach nearly perfect performance, as shown in Table 7. The R/R modality, in which the images are rotated by a random rotation in $\mathrm{SO}(3)$, is specifically designed for testing equivariant models. Therefore, these have a substantial advantage since they do not need to learn the symmetry of the task. In the R/R modality, our model is only outperformed by some equivariant models and performs better than or on par with all other models. Note, however, that the equivariant models do not scale to high-resolution inputs.

C. Experimental details

C.1. SynWoodscape and MNIST experiments

In Table 8 we provide further details on the spatial size of the features throughout the HEAL-SWIN model used in the experiments discussed in Section 4.

Resolution In order to eliminate resolution as a central parameter in comparing the HEAL-SWIN to the SWIN, we first rescale the input images to a size of 640×768 giving a resolution of

[5] https://drive.google.com/drive/folders/1N5rrySiw1uh9kLeBuOblMbXJ09YsqO7I

[6] http://buildingparser.stanford.edu/dataset.html

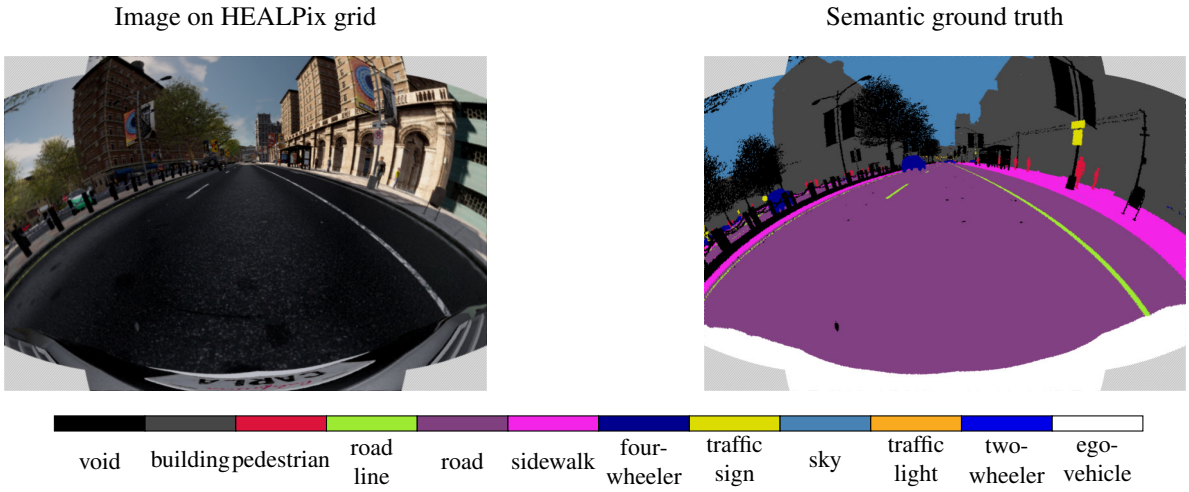


Figure 8. Sample RGB image (left) and semantic segmentation ground truth (right) from the Large+AD SynWoodScape dataset, projected onto the plane for visualization. Regions not covered by the $^{8}/_{12}$ base pixels of the HEALPix grid are hatched.

approximately 492k, which we can sample to the HEALPix grid using $n_{\text{side}} = 256$ yielding a resolution of around ~ 525k.

Hardware and training details For the semantic segmentation task we train all models on four `Nvidia A40` GPUs with an effective batch size of 8 and a constant learning rate of 9.4×10^{-4}. For the depth estimation task we used an effective batch size of 4 and learning rates of 5×10^{-3} and 5×10^{-5} for the HEAL-SWIN and SWIN models, respectively, chosen from the best performing models after a learning rate ablation.

Classes and reweighting We adjust the number of output channels in the base HEAL-SWIN and SWIN models described in Section 4 to the number of classes and train with a weighted pixel-wise cross-entropy loss. We choose the class weights w_i to be given in terms of the class prevalences n_i by $w_i = n_i^{-1/4}$.

HEAL-SWIN versus SWIN for flat segmentation In Table 12, we show the results of evaluating the segmentation models discussed in Section 4.1 on the plane. In this case, the HEAL-SWIN predictions are projected onto the pixel grid of the SWIN predictions before evaluation. To ensure a fair comparison, the flat mIoU is calculated on a masked region of this grid, removing pixels which lie outside of the (restricted) HEALPix grid we use.

C.2. Stanford 2D-3D-S experiments

Resolution In order to be close to the resolution used by HexRUNet [50], we chose $n_{\text{side}} = 64$ which resulted in 49k pixels in the HEALPix grid, compared to 20k pixels used for HexRUNet.

Hardware and training details Training was conducted with a constant learning rate of 5×10^{-3} on four `Nvidia A100` due to the smaller demand on them compared to the `Nvidia A40` on the compute cluster, although the `A40`'s would work perfectly fine. The experiments on the Stanford 2D-3D-S dataset used an effective batch size of 80, a small weight decay of 0.1 and a gradient clipping on 0.5 acting on the total gradient 2-norm.

Additional results For a per-class performance breakdown comparing the HEAL-SWIN model to previous comparable models, see Table 5 and 6.

In Figure 12 and 13 we show the best and worst predictions of our model respectively.

Model architecture For a good comparison to HexRUNet [50] and UGSCNN [29], which have 1.5M and 5.2M parameters, respectively, we construct a HEAL-SWIN model with 1.5M parameters. Table 10 shows the spatial features per block. We performed ablations to set window, patch, and shift sizes, see Table 9.

Computational complexity To verify the limited computational overhead of the HEAL-SWIN architecture we provide ablations over a range of resolutions in Table 11.

Table 4. Classes from the Large SynWoodScape and the Large+AD SynWoodScape datasets in terms of the classes provided by SynWoodScape.

SynWoodScape	Our Classes	
	Large SynWoodScape	Large+AD SynWoodScape
unlabeled	void	void
building	building	building
fence	void	void
other	void	void
pedestrian	void	pedestrian
pole	void	void
road line	road line	road line
road	road	road
sidewalk	sidewalk	sidewalk
vegetation	void	void
four-wheeler vehicle	four-wheeler vehicle	four-wheeler vehicle
wall	void	void
traffic sign	void	traffic sign
sky	sky	sky
ground	void	void
bridge	void	void
rail track	void	void
guard rail	void	void
traffic light	void	traffic light
water	void	void
terrain	void	void
two-wheeler vehicle	void	two-wheeler vehicle
static	void	void
dynamic	void	void
ego-vehicle	ego-vehicle	ego-vehicle

Table 5. Per-class intersection over union of spherical models on the Stanford 2D-3D dataset. The same instance of HEAL-SWIN is evaluated for the cases where the polar ground truth is kept (†) and where it is mapped to background (∗).

Method	mIoU	beam	board	bookcase	ceiling	chair	clutter	column	door	floor	sofa	table	wall	window
Gauge CNN [9]	39.4	–	–	–	–	–	–	–	–	–	–	–	–	–
UGSCNN [29]	38.3	8.7	32.7	33.4	82.2	42.0	25.6	10.1	41.6	87.0	7.6	41.7	61.7	23.5
HexRUNet [50]	43.3	10.9	39.7	37.2	**84.8**	50.5	29.2	11.5	45.3	**92.9**	19.1	49.1	63.8	29.4
SphCNN [18, 19]	40.2	-	-	-	-	-	-	-	-	-	-	-	-	-
Spin-SphCNN [19]	41.9	-	-	-	-	-	-	-	-	-	-	-	-	-
HEAL-SWIN*	44.3	**11.8**	**42.8**	**42.0**	67.2	57.8	**33.9**	**12.9**	50.9	66.0	**24.5**	**56.8**	68.7	**40.2**
HEAL-SWIN†	**47.3**	11.5	42.8	42.0	83.8	**58.8**	33.8	12.8	**52.0**	87.6	24.4	56.8	**68.8**	40.2

Table 6. Per-class accuracy of spherical models on the Stanford 2D-3D dataset.

Method	mAcc	beam	board	bookcase	ceiling	chair	clutter	column	door	floor	sofa	table	wall	window
Gauge CNN [9]	55.9	–	–	–	–	–	–	–	–	–	–	–	–	–
UGSCNN [29]	54.7	19.6	48.6	49.6	93.6	63.8	43.1	**28.0**	63.2	96.4	21.0	70.0	74.6	39.0
HexRUNet [50]	58.6	**23.2**	56.5	**62.1**	94.6	66.7	41.5	18.3	64.5	96.2	41.1	**79.7**	77.2	41.1
SphCNN [18, 19]	52.8	-	-	-	-	-	-	-	-	-	-	-	-	-
Spin-SphCNN [19]	55.6	-	-	-	-	-	-	-	-	-	-	-	-	-
HEAL-SWIN	**61.9**	18.9	**58.3**	61.0	**95.6**	**75.4**	**50.9**	20.2	**66.5**	**97.7**	**41.3**	76.7	**88.9**	**52.7**

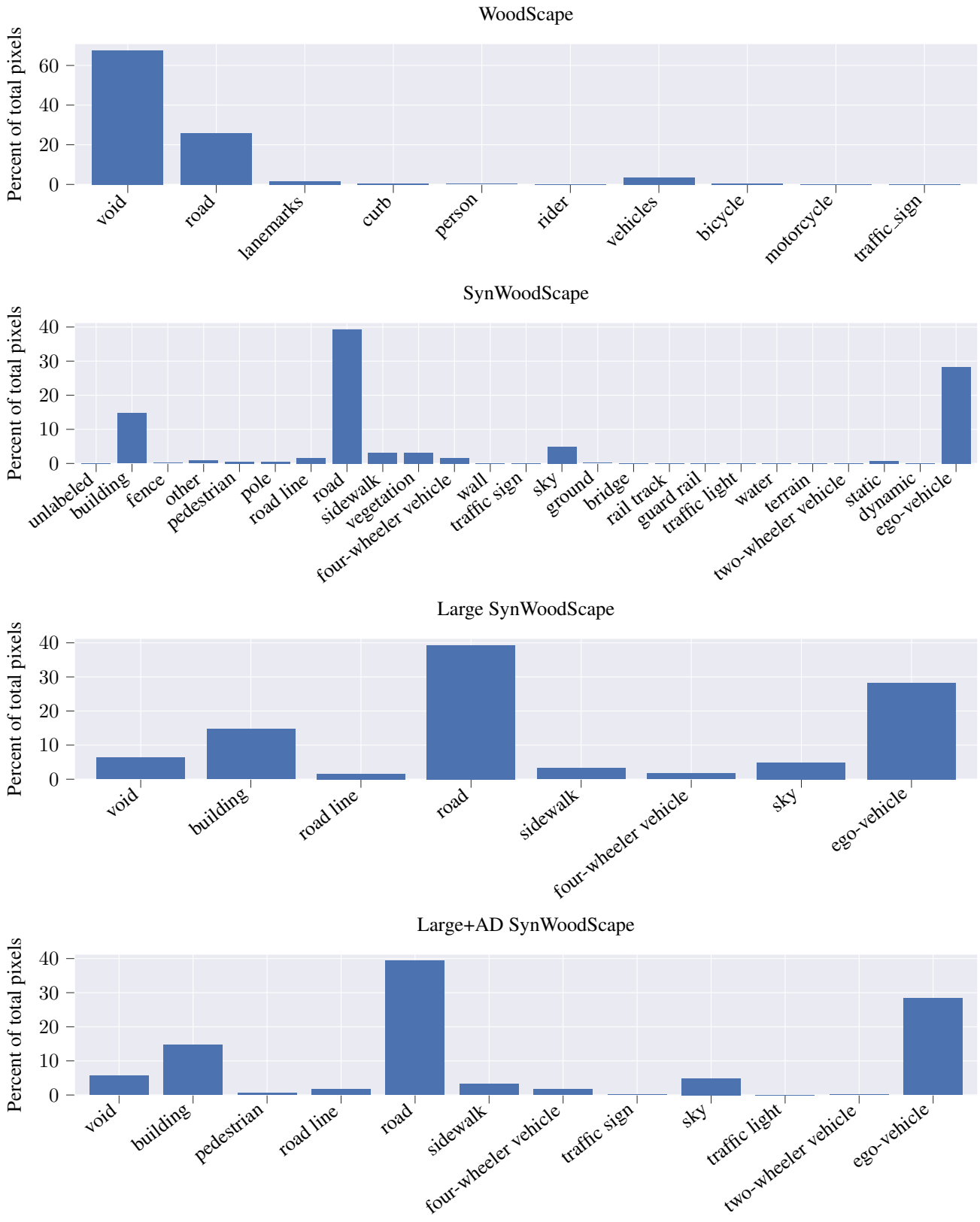
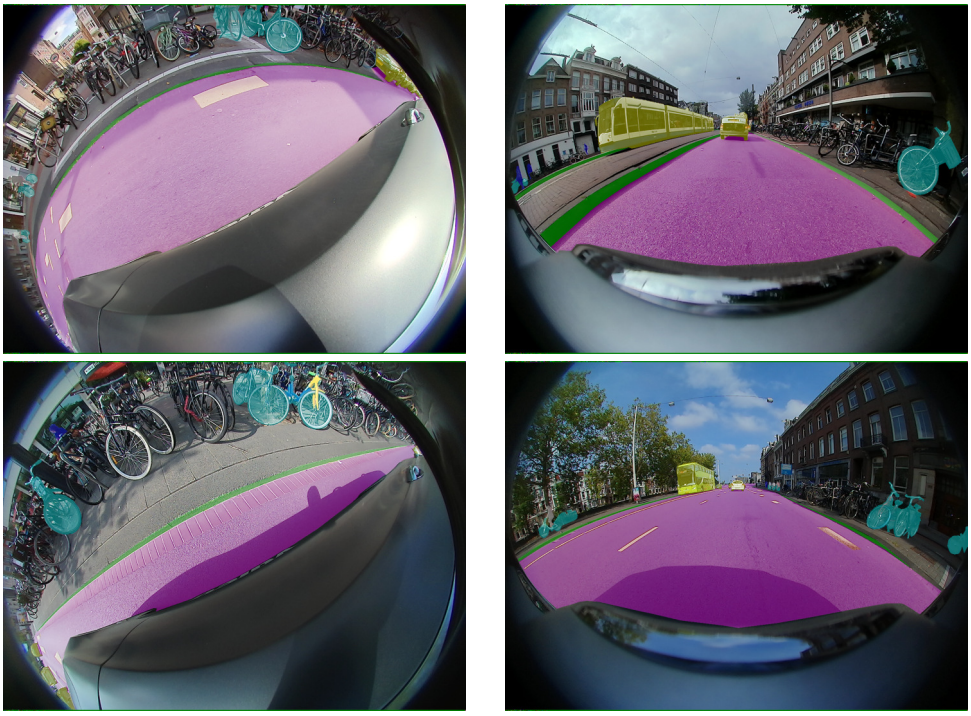
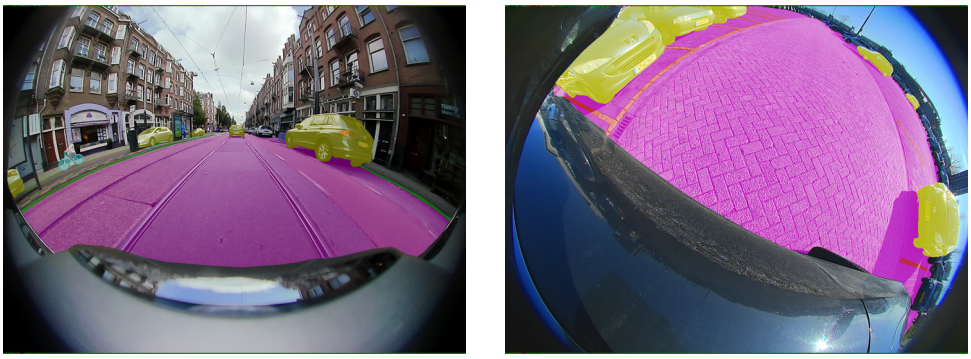


Figure 9. Class distributions for the datasets used in semantic segmentation.

(a) Large parts of the ego vehicle are labeled as *lanemarks*.

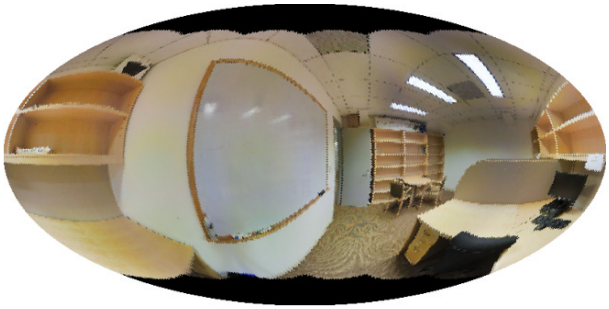


(b) Some (but not all) parked bicycles are labeled as *bicycle*.

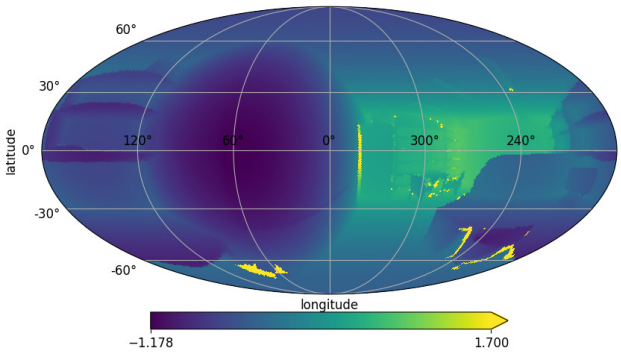


(c) Some (but not all) parked cars are labeled as *vehicles*.

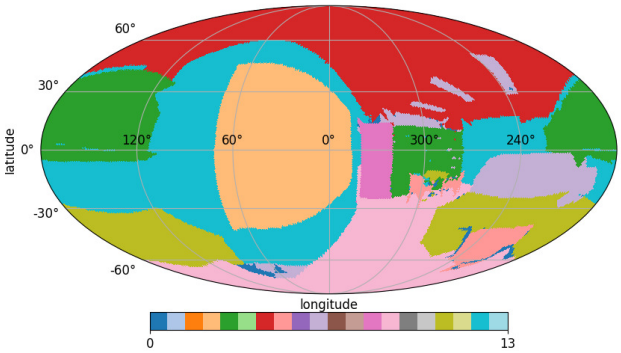
Figure 10. Examples of inconsistencies in semantic masks of the WoodScape dataset.



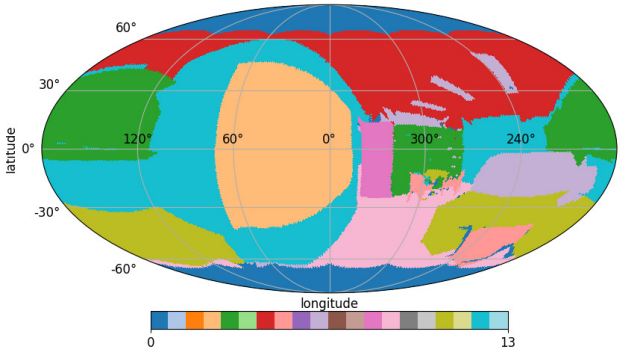
(a) The RGB channels. For the polar regions all RGB values are zero and hence here shown in black.



(b) Log of the depth channel. Unknown depth information is represented as values above $\sim$ 65m, in the figure shown as solid yellow. Those areas are also prescribed the unknown class in the ground truth (mapped to background during training) which are shown in dark blue in Subfigure (c). Note that the polar regions have valid depth values.



(c) Full semantic ground truth. Note that the polar regions have full semantic ground truth even though the RGB channels lack information in these areas.



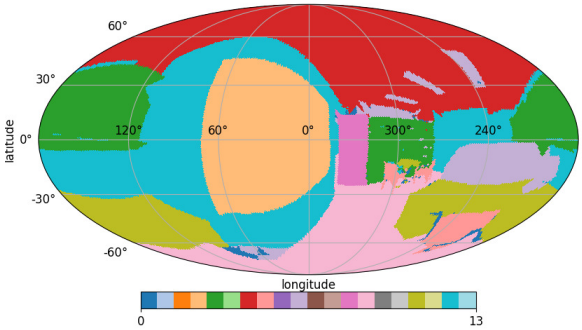
(d) Semantic ground truth with the polar regions mapped to background. This is what the model is trained on.

Figure 11. Visualisation of a RGBD sample from the Stanford 2D-3D-S dataset. Subfigure (a) shows the RGB channels while Subfigure (b) displays the depth channels, where areas with unknown depth are shown in solid yellow. These together form the input to the network. Subfigure (c) shows the full semantic segmentation ground truth with ground truth also in the polar regions. Note that the areas corresponding to background/the unknown class, shown in dark blue, are the same areas that have unknown depth in Subfigure (b). Subfigure (d) shows the semantic ground truth after the polar regions have been mapped to background which is what the model is trained on.

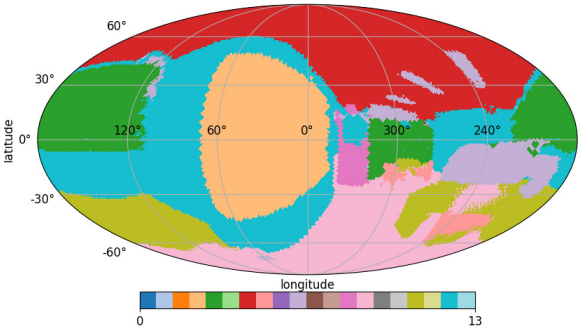
Table 7. Classification accuracy on spherical MNIST when trained and evaluated on non-rotated data (NR/NR) and on rotated data (R/R). Equivariant models are marked with an asterisk.

Model	NR/NR Acc	R/R Acc
S2CNN* [10]	96	95
Clebsch-Gordan Nets* [30]	96.4	96.6
Gauge CNN* [9]	99.43	**99.31**
SphCNN* [18, 19]	98.75	98.71
Spherical Transformer* [7]	–	95.09
UGSCNN [29]	99.23	94.92
HexRUnet [50]	**99.45**	97.05
HEAL-SWIN (Ours)	99.20	96.96

(a) The RGB channels.



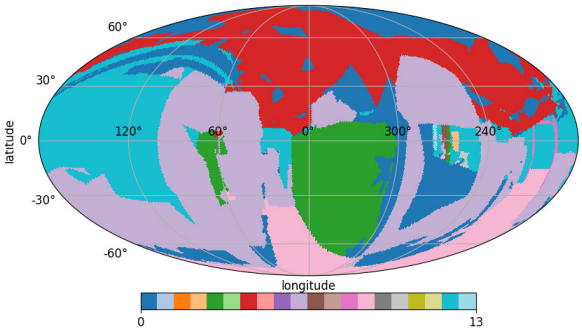
(b) The semantic ground truth.



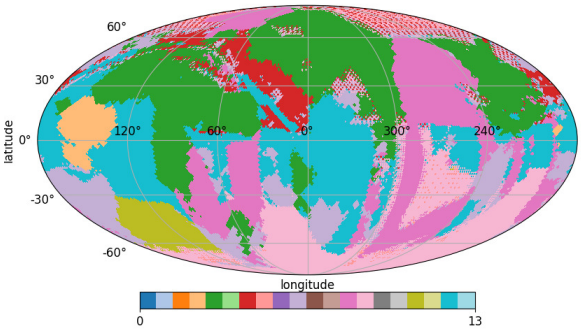
(c) The predicted segmentation.

Figure 12. One of the best per sample predictions. Recall that during training the model did not update on areas lacking RGB data. This sample is from the evaluation set in cross validation fold 1.

(a) The RGB channels. Note that this sample is very visually noisy.



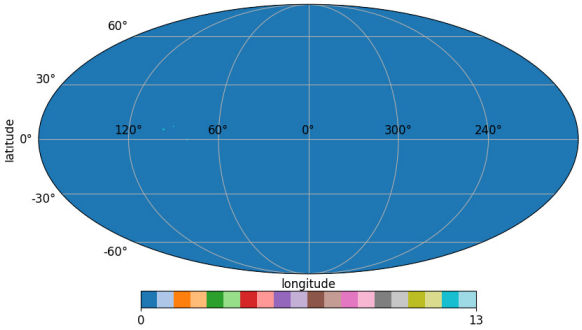
(b) The semantic ground truth. Note that large irregular areas are unknown (and hence mapped to background).



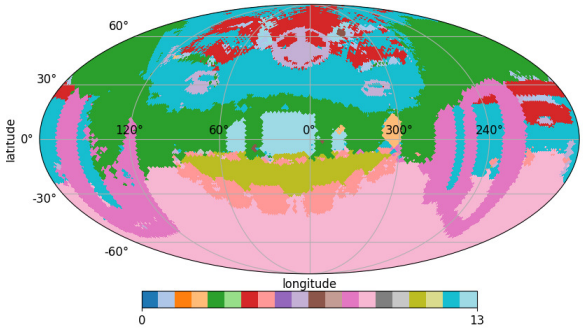
(c) The predicted segmentation.

Figure 13. One of the worst per sample predictions. Recall that during training the model did not update on areas lacking RGB data. This sample is from the evaluation set in cross validation fold 1.

(a) RGB channels.



(b) Ground truth.



(c) Predicted segmentation.

Figure 14. The worst per sample prediction. Note that the predicted segmentation is reasonable but that the ground truth, for some reason, consists of only the background class. This sample is from the evaluation set in cross validation fold 3. Although neither this nor similar samples were removed from the dataset during training, they were effectively ignored since the background class is ignored during training.

Table 8. Spatial features per layer in the HEAL-SWIN models used for the experiments described in Section 4.

layer	pixel / patches	windows	windows per base pixel	n_{side}	followed by
input	524288	8192	1024	256	patch embedding
HEAL-SWIN block 1	131072	2048	256	128	patch merging
HEAL-SWIN block 2	32768	512	64	64	patch merging
HEAL-SWIN block 3	8192	128	16	32	patch merging
HEAL-SWIN block 4	2048	32	4	16	patch expansion
HEAL-SWIN block 5	8192	128	16	32	patch expansion
HEAL-SWIN block 6	32768	512	64	64	patch expansion
HEAL-SWIN block 7	131072	2048	256	128	patch expansion
output	524288	8192	1024	256	—

Table 9. Ablations over patch size, window size, shift size and shifting strategy on the Stanford 2D-3D. Performance is measured using the three-fold cross-validation of that dataset. Unless otherwise stated, the model parameters are $n_{\text{patch}} = 4$, $n_{\text{win}} = 64$, spiral shifting with $n_{\text{shift}} = 4$ and the model architecture is the same whose performance is reported in Table 3.

Parameter values		mIoU	mAcc
$n_{\text{patch}} = 4$		43.2	61.1
$n_{\text{patch}} = 16$		40.9	58.9
$n_{\text{win}} = 64$	$n_{\text{shift}} = 4$	43.2	61.1
$n_{\text{win}} = 16$	$n_{\text{shift}} = 2$	44.3	61.9
$n_{\text{shift}} = 2$		42.2	60.5
$n_{\text{shift}} = 4$		43.2	61.1
$n_{\text{shift}} = 8$		39.3	56.6
spiral shifting		43.2	61.1
grid shifting		42.3	59.9

Table 10. Spatial features per layer in the HEAL-SWIN models used for the Stanford 2D3Ds experiments.

layer	pixel / patches	windows	windows per base pixel	n_{side}	followed by
input	49152	1536	256	64	patch embedding
HEAL-SWIN block 1	12288	768	64	32	patch merging
HEAL-SWIN block 2	3072	192	16	16	patch merging
HEAL-SWIN block 3	768	48	4	8	patch expansion
HEAL-SWIN block 4	3072	192	16	16	patch expansion
HEAL-SWIN block 5	12288	768	64	32	patch expansion
output	49152	1536	256	64	—

Table 11. Comparison of inference times for HEAL-SWIN and SWIN ablated over data resolution.

	Resolution	Pixels	time / pixel
HEAL-SWIN	8×256.0^2	$5.2 \cdot 10^5$	297 ± 26ns
SWIN	640×768	$4.9 \cdot 10^5$	296 ± 39ns
HEAL-SWIN	8×128.0^2	$1.3 \cdot 10^5$	559 ± 127ns
SWIN	256×384	$1.0 \cdot 10^5$	660 ± 171ns
HEAL-SWIN	8×64.0^2	$0.3 \cdot 10^5$	2031 ± 519ns
SWIN	128×160	$0.2 \cdot 10^5$	2792 ± 668ns

Table 12. Mean IoU for semantic segmentation with HEAL-SWIN and SWIN, averaged over three runs, after projection onto the plane. For WoodScape, we exclude the *void* class from the mean but keep it in the loss.

Model	Dataset	flat mIoU
HEAL-SWIN	Large SynWoodScape	0.899
SWIN	Large SynWoodScape	0.930
HEAL-SWIN	Large+AD SynWoodScape	0.790
SWIN	Large+AD SynWoodScape	0.837
HEAL-SWIN	WoodScape	0.611
SWIN	WoodScape	0.620