 Supplementary Material  
HoHoNet: 360 Indoor Holistic Understanding with Latent Horizontal Features

A. Network architecture diagram

We show the detailed architecture diagram in Fig. A. The shape of each feature tensor is denoted as “# of channels, height, width” within the box. The height and width of the input panorama are assumed to be 512 and 1024 respectively. D and E are hyperparameters. The ConvSqueezeH layer is a depthwise convolution layer with kernel size set to the prior known input feature height without padding, which produces output feature height 1.

![Network Architecture Diagram](image)

Figure A: The detailed network architecture with ResNet50 [4] backbone.

B. Comparing EHC block and HC block [8]

The height compression block aims to squeeze a 2D feature from the backbone to produce a 1D horizontal feature. Fig. B shows the architecture of our Efficient Height Compression block (EHC block) and the one of HC block [8] for comparison. The HC block [8] employs a sequence of convolution layers to gradually reduce the number of channels and heights, while we first use a convolution layer for channel reduction and then use bilinear upsampling and ConvSqueezeH layer to produce the features in horizontal shape. We show in our ablation experiments that replacing the HC block [8] with the proposed ECH block leads to better speed and accuracy.

![Comparison of EHC and HC blocks](image)

Figure B: Comparison of the proposed EHC block and the HC block in [8].
C. Detailed layout estimation results

We show detailed quantitative results for room layout under different numbers of ground truth 2D corners in Table A. Our training protocol and layout formalization are identical to HorizonNet [8], while we observe improvements (except rooms with six corners) by using our network architecture. In comparison with the most recent state-of-the-art—AtlantaNet [7], we show better results on scenes with fewer corners and similar accuracy on overall scenes; meanwhile, our model is 22× faster than AtlantaNet [7]. For depth-based evaluation proposed by LayoutNet v2 [13], we use an in-house implementation to synthesize layout depth as the ground truth layout depth is not available from the MatterportLayout dataset and the provided implementation for synthesizing layout depth produces invalid values (i.e., zero) for some cases. Note that our implementation is very different to the [13]’s implementation so the results are not directly comparable. LayoutNet v2 [13], AtlantaNet [7] and our method achieve 0.28, 0.20 and 0.22 RMSE respectively; 0.90, 0.94 and 0.95 9x respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th># of corners</th>
<th>3D IoU (%)</th>
<th>2D IoU (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>overall</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>LayoutNet v2  [13]</td>
<td>75.82</td>
<td>81.35</td>
<td>72.33</td>
</tr>
<tr>
<td>DuLa-Net v2    [10]</td>
<td>75.07</td>
<td>77.02</td>
<td>78.79</td>
</tr>
<tr>
<td>HorizonNet     [8]</td>
<td>79.11</td>
<td>81.88</td>
<td>82.26</td>
</tr>
<tr>
<td>AtlantaNet     [7]</td>
<td><strong>80.02</strong></td>
<td>82.09</td>
<td>82.08</td>
</tr>
<tr>
<td>Ours</td>
<td>79.88</td>
<td><strong>82.64</strong></td>
<td>82.16</td>
</tr>
</tbody>
</table>

Table A: Detailed quantitative comparison for room layout estimation on MatterportLayout [13] under different numbers of ground-truth corners.

D. Detailed semantic segmentation results

We show detailed per-class IoU and per-class Acc for semantic segmentation in Table B. We achieve the best IoU on 10 out of 13 classes and superior overall mIoU; we achieve best Acc on 7 out of 13 classes and comparable overall mAcc.

<table>
<thead>
<tr>
<th>Method</th>
<th># of corners</th>
<th>overall</th>
<th>beam</th>
<th>board</th>
<th>bookcase</th>
<th>ceiling</th>
<th>chair</th>
<th>clutter</th>
<th>column</th>
<th>door</th>
<th>floor</th>
<th>sofa</th>
<th>table</th>
<th>wall</th>
<th>window</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>overall</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>10+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UGSCNN [5]</td>
<td>38.3</td>
<td>8.7</td>
<td>32.7</td>
<td>33.4</td>
<td>82.2</td>
<td>42.0</td>
<td>25.6</td>
<td>10.1</td>
<td>41.6</td>
<td>87.0</td>
<td>7.6</td>
<td>41.7</td>
<td>61.7</td>
<td>23.5</td>
<td></td>
</tr>
<tr>
<td>HexRUNet [12]</td>
<td>43.3</td>
<td><strong>10.9</strong></td>
<td>39.7</td>
<td>37.2</td>
<td>84.8</td>
<td>50.5</td>
<td>29.2</td>
<td>11.5</td>
<td>45.3</td>
<td>92.9</td>
<td>19.1</td>
<td>49.1</td>
<td>63.8</td>
<td>29.4</td>
<td></td>
</tr>
<tr>
<td>TangentImg [3]</td>
<td>37.5</td>
<td>10.9</td>
<td>26.6</td>
<td>31.9</td>
<td>82.0</td>
<td>38.5</td>
<td>29.3</td>
<td>5.9</td>
<td>36.2</td>
<td>89.4</td>
<td>12.6</td>
<td>40.4</td>
<td>56.5</td>
<td>26.7</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>40.8</td>
<td>3.6</td>
<td>43.5</td>
<td>40.6</td>
<td>81.8</td>
<td>41.3</td>
<td>27.7</td>
<td>9.2</td>
<td>52.0</td>
<td>92.2</td>
<td>9.4</td>
<td>44.6</td>
<td>61.6</td>
<td>23.4</td>
<td></td>
</tr>
</tbody>
</table>

Table B: Detailed quantitative per-class results on Stanford2D3D [1] with RGB-D as input.
E. More qualitative comparisons for depth estimation

We show more qualitative comparisons with the prior art—BiFuse [9]—in Fig. C. BiFuse’s results are obtained from their official released model trained on the real-world Matterport3D [2] dataset.

Figure C: More qualitative comparisons of the estimated dense depth with the prior art—BiFuse [9]. The ‘Advantage’ column shows the MAE difference between ours and BiFuse’s where the blue color indicates ours is better and the red color for vice versa.
F. Qualitative results for semantic segmentation

Qualitative results for semantic segmentation on Stanford2D3D [1] dataset are shown in Fig. D. We fail to build the prior art [3] from their public release for semantic segmentation on high-resolution panorama, so we only show our results.

Figure D: Qualitative results for semantic segmentation on Stanford2D3D [1] dataset.
G. Qualitative comparisons for layout estimation

We show qualitative comparisons for room layout estimation with the prior art—AtlantaNet [7]—in Fig. E. The results of AtlantaNet are obtained from their official code and pre-trained weights. We use [8] post-processing algorithm to produce Manhattan layouts; AtlantaNet [7]'s algorithm generates less restrictive Atlanta layouts. Our model achieves promising results comparable to the most recent AtlantaNet [7], while our model runs $22\times$ faster.

Figure E: Qualitative comparisons for room layout estimated with the competitive AtlantaNet [7]. The green, magenta, and blue are the ground truth layout, AtlantaNet's results, and our results respectively.
H. 3D visualization for the estimated depth

Figure F: 3D visualization for the estimated depth by HoHoNet.

I. 3D visualization for the estimated layout

Figure G: 3D visualization for the estimated layout by HoHoNet.

J. Future work

There are plenty of potential future directions upon the proposed framework. (i) In this work, we only present two generic operations—linear interpolation and the inverse discrete cosine transform—to predict dense from the LHFeat. With the unified basis view between the two specific operations, we hope future work can find an even better basis or task-specialized basis. (ii) Developing the ERP distortion-aware technique upon our framework could also be more uncomplicated as our “decoder” is horizontal, enabling future work to focus only on the backbone layers. (iii) The effectiveness of the proposed HoHoNet is only showcased on each modality separately in this work. Extend the recent works on 360° multi-modalities regularization [6] or 360° cascade multi-stages modeling [11] upon the LHFeat is a potential direction. (iv) Finally, despite the generally good performance of HoHoNet, our visualization reveals the weakness of HoHoNet on the boundary region and high-frequency signal in a column. Future work on proposing a remedy for the observed issue is desirable.
References


