ShapeConv Supplementary Material

Jinming Cao[1] Hanchao Leng[1] Dani Lischinski[2] Danny Cohen-Or[3] Changhe Tu[1*] Yangyan Li[4*]
[1]Shandong University, China [2]The Hebrew University of Jerusalem, Israel
[3]Tel Aviv University, Israel [4]Alibaba Group, China
{jinming.ccao, hanchao.leng, danix3d, cohenor, changhe.tu, yangyan.lee}@gmail.com

1. Detailed Results

1.1. More Qualitative Results

Figure 1 illustrates more qualitative results on NYUDv2-13 and -40. The depth information, especially the detailed one, can be well utilized by ShapeConv to extract the object features, like the chair in Figure 1(a), the lamp in Figure 1(b), the cabinets in Figure 1(c) and the faucet in Figure 1(e), The key observations from this figure are as follows. Firstly, ShapeConv can significantly improve the segmentation results in edge areas compared with the baseline. Secondly, ShapeConv produce a positive effect on image regions with weak lightness, such as the example in Figure 1(d).

1.2. More Ablation Results

Table 1. Ablation study on the NYUDv2-13 dataset.

Back bone	Setting	Pixel Acc.(%)	Mean Acc.(%)	Mean IoU.(%)	f.w. IoU.(%)
Res Net -101	a.RGB	78.3	71.1	57.9	65.0
	b.RGB*	79.3	71.8	59.1	66.3
	c.RGB+Depth	80.4	73.6	61.1	68.1
	d.RGB+Depth*	81.2	74.1	62.3	69.1
	e.RGB+HHA	80.0	73.4	61.3	67.6
	f.RGB+HHA*	81.0	74.3	63.1	68.9
	g.RGB+Depth+ShapeConv	81.3	73.8	62.5	69.3
	h.RGB+Depth+ShapeConv*	81.9	74.5	63.5	70.1
	i.RGB+HHA+ShapeConv	81.2	74.9	62.9	69.1
	j.RGB+HHA+ShapeConv*	81.9	75.7	64.0	70.1
Res Net -50	a.RGB	77.5	69.3	56.2	64.1
	b.RGB*	78.3	69.9	57.3	65.1
	c.RGB+Depth	79.5	72.6	60.1	66.9
	d.RGB+Depth*	80.3	73.3	61.3	68.0
	e.RGB+HHA	80.0	72.5	60.8	67.6
	f.RGB+HHA*	80.6	72.7	61.6	68.4
	g.RGB+Depth+ShapeConv	80.4	72.6	61.2	68.0
	h.RGB+Depth+ShapeConv*	81.3	73.3	62.6	69.2
	i.RGB+HHA+ShapeConv	80.4	73.0	61.8	68.1
	j.RGB+HHA+ShapeConv*	81.1	73.4	62.7	69.1

Table 2. Ablation study on the NYUDv2-40 dataset.

Back bone	Setting	Pixel Acc.(%)	Mean Acc.(%)	Mean IoU.(%)	f.w. IoU.(%)
Res Net -101	a.RGB	71.8	56.9	43.9	57.3
	b.RGB*	72.8	57.8	45.3	58.2
	c.RGB+Depth	72.8	58.9	44.9	57.7
	d.RGB+Depth*	73.9	59.1	46.8	60.0
	e.RGB+HHA	73.4	58.9	45.9	59.7
	f.RGB+HHA*	74.4	60.2	47.6	60.7
	g.RGB+Depth+ShapeConv	73.9	58.2	46.2	60.0
	h.RGB+Depth+ShapeConv*	74.8	59.2	47.5	60.8
	i.RGB+HHA+ShapeConv	74.5	59.5	47.4	60.8
	j.RGB+HHA+ShapeConv*	75.5	60.7	49.0	61.7
Res Net -50	a.RGB	70.8	55.2	42.5	56.2
	b.RGB*	71.8	56.3	43.9	57.0
	c.RGB+Depth	72.1	56.4	44.3	58.0
	d.RGB+Depth*	73.2	57.5	45.7	58.9
	e.RGB+HHA	73.1	57.7	45.6	59.2
	f.RGB+HHA*	74.2	59.0	47.1	60.2
	g.RGB+Depth+ShapeConv	72.9	56.4	45.1	58.6
	h.RGB+Depth+ShapeConv*	74.1	57.9	46.7	59.8
	i.RGB+HHA+ShapeConv	74.1	59.1	47.3	60.5
	j.RGB+HHA+ShapeConv*	75.0	60.4	48.8	61.4

To provide a more in-depth analysis of ShapeConv, we conducted detailed ablation studies on the NYUDv2 dataset with deeplabv3+ as baseline, and show the results of NYUDv2-13 and -40 in Table 1 and Table 2, respectively, the RGB, Detph and HHA in table denote the inputs consisting of RGB images, depth images and HHA images, respectively. In these two tables, two backbones are further utilized, i.e., ResNet-50 and -101. The key observations are consistent with those reported in the main manuscript.

2. Implementation Details

We implemented our network using the publicly available Pytorch[1] [1]. We used the SGD [2] optimizer and set the momentum and weight decay to 0.9 and 0.0001, respectively. The batch size is 4 for NYUDv2 and SUN-RGBD dataset, 2

*Corresponding Author
[1]https://pytorch.org/

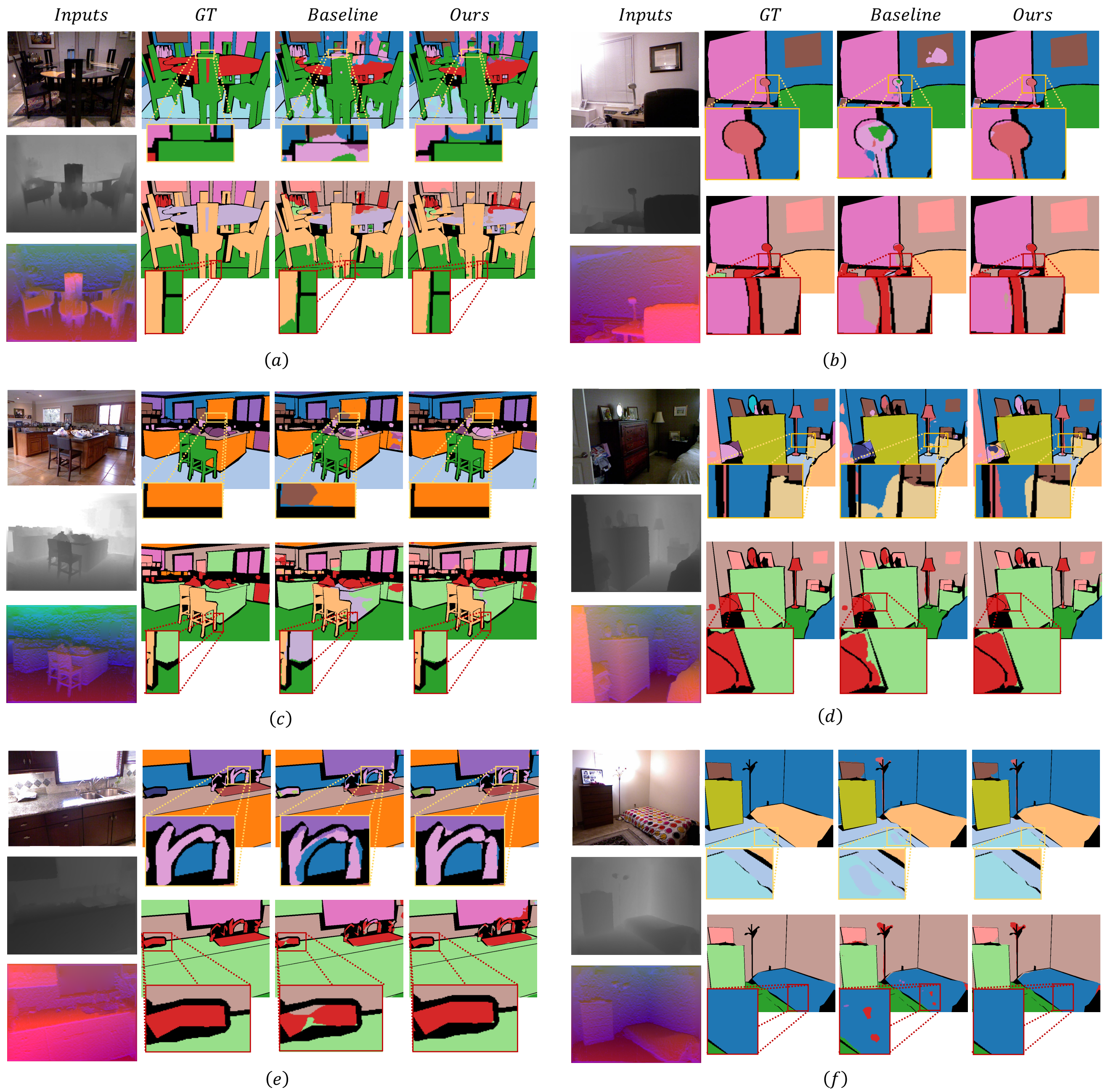

Figure 1. Visualization results from NYUDv2 dataset. Input column denotes RGB, Depth, HHA images from top to bottom; the black regions in the GT, Baseline and Ours indicate the ignored category. The upper and lower cases are from NYUDv2-40 and NYUDv2-13, respectively.

for SID dataset. We used an initial learning rate of 0.007 and decay it to 0.002. The code is available and the guideline for reproducing the results can be found at https://github.com/hanchaoleng/ShapeConv.

3. Proof for Equation 6

The two formulations of ShapeConv in Equation 2 and Equation 5 are mathematically equivalent, i.e.,

$$\mathbb{F} = ShapeConv(\mathbb{K}, \mathbb{W}_B, \mathbb{W}_S, \mathbb{P}) = Conv(\mathbb{K}, \mathbf{P_{BS}}) = Conv(\mathbf{K_{BS}}, \mathbb{P})$$

Proof.

$$
\begin{aligned}
\mathbb{F}_{c_{out}} = & \sum_{k}^{K_h \times K_w \times C_{in}} (\mathbb{K}_{k,c_{out}} \times \mathbf{P}_{\mathbf{BS}_k}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbb{K}_{i,j,c_{out}} \times \mathbf{P}_{\mathbf{BS}_{i,j}}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbb{K}_{i,j,c_{out}} \times (\mathbf{P}_{\mathbf{B}_{1,j}} + \mathbf{P}_{\mathbf{S}_{i,j}})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbb{K}_{i,j,c_{out}} \times (\mathbb{W}_B \times \mathbb{P}_{B_{1,j}} + \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{P}_{S_{m,j}})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbb{W}_B \times \mathbb{K}_{i,j,c_{out}} \times \mathbb{P}_{B_{1,j}} + \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,c_{out}} \times \mathbb{P}_{S_{m,j}})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbb{W}_B \times \mathbb{K}_{B_{1,j,c_{out}}} \times \mathbb{P}_{i,j} + \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,c_{out}} \times (\mathbb{P}_{m,j} - \mathbb{P}_{B_{1,j}})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j}) + \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,c_{out}} \times \mathbb{P}_{m,j} - \mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,c_{out}} \times \mathbb{P}_{B_{1,j}})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j}) + \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{i,j,c_{out}} \times \mathbb{P}_{m,j} - \mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{B_{1,j,c_{out}}} \times \mathbb{P}_{m,j})) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j}) + \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times (\mathbb{K}_{i,j,c_{out}} - \mathbb{K}_{B_{1,j,c_{out}}}) \times \mathbb{P}_{m,j}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j}) + \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} \sum_{m}^{K_h \times K_w} (\mathbb{W}_{S_{m,i,j}} \times \mathbb{K}_{S_{i,j,c_{out}}} \times \mathbb{P}_{m,j}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j}) + \sum_{m}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{S}_{m,j,c_{out}}} \times \mathbb{P}_{m,j}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} \times \mathbb{P}_{i,j} + \mathbf{K}_{\mathbf{S}_{i,j,c_{out}}} \times \mathbb{P}_{i,j}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} ((\mathbf{K}_{\mathbf{B}_{1,j,c_{out}}} + \mathbf{K}_{\mathbf{S}_{i,j,c_{out}}}) \times \mathbb{P}_{i,j}) \\
= & \sum_{i}^{K_h \times K_w} \sum_{j}^{C_{in}} (\mathbf{K}_{\mathbf{BS}_{i,j,c_{out}}} \times \mathbb{P}_{i,j}) \\
= & \sum_{k}^{K_h \times K_w \times C_{in}} (\mathbf{K}_{\mathbf{BS}_{k,c_{out}}} \times \mathbb{P}_k)
\end{aligned}
$$

References

[1] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[2] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.