Supplementary Material for Pyramid Multi-view Stereo Net with Self-adaptive View Aggregation

Hongwei Yi[1*], Zizhuang Wei[1*], Mingyu Ding[2], Runze Zhang[3], Yisong Chen[1], Guoping Wang[1], and Yu-Wing Tai[4]

[1] PKU {hongweiyi, weizizhuang, chenyisong, wgp}@pku.edu.cn
[2] HKU mying@cs.hku.hk
[3] Tencent ryanrzzhang@tencent.com
[4] Kwai Inc. yuwing@gmail.com

1 Network Architecture

The network details of **2D U-Net** and **3D U-Net** in **VA-MVSNet** are described in Table 1.

2 More Reconstruction Results

This section presents all the reconstructions results on *DTU* [1] *evaluation* dataset in Fig. 1 and *Tanks and Temples* [20] in Fig. 2. PVA-MVSNet is able to reconstruct dense, accurate and complete point clouds on *DTU* [1] dataset and shows strong generalization on all scenes in *Tanks and Temples* [20].

Fig. 1. Reconstruction results on DTU [1] evaluation set.

* Equal Contribution

Input	Layer Destription	Output	Output Size
Input multi-view image size: $N \times H \times W \times 3$			
2D U-Net			
$I_{i=0 \cdots N-1}$	ConvGR, filter=3×3, stride=2	2D1_0	$H \times W \times 16$
2D0_0	ConvGR, filter=3×3, stride=2	2D2_0	$H \times W \times 32$
2D0_1	ConvGR, filter=3×3, stride=2	2D3_0	$H \times W \times 64$
2D0_2	ConvGR, filter=3×3, stride=2	2D4_0	$H \times W \times 128$
$I_{i=0 \cdots N-1}$	ConvGR, filter=3×3	2D0_1	$H \times W \times 8$
2D0_1	ConvGR, filter=3×3	2D0_2	$H \times W \times 8$
2D1_0	ConvGR, filter=3×3	2D1_1	$H \times W \times 16$
2D1_1	ConvGR, filter=3×3	2D1_2	$H \times W \times 16$
2D2_0	ConvGR, filter=3×3	2D2_1	$H \times W \times 32$
2D2_1	ConvGR, filter=3×3	2D2_2	$H \times W \times 32$
2D3_0	ConvGR, filter=3×3	2D3_1	$H \times W \times 64$
2D3_1	ConvGR, filter=3×3	2D3_2	$H \times W \times 64$
2D4_0	ConvGR, filter=3×3	2D4_1	$H \times W \times 128$
2D4_1	ConvGR, filter=3×3	2D4_2	$H \times W \times 128$
2D4_2	DeConvGR, filter=3×3, stride=2	2D5_0	$H \times W \times 64$
[2D5_0, 2D3_2]	ConvGR, filter=3×3	2D5_1	$H \times W \times 64$
2D5_1	ConvGR, filter=3×3	2D5_2	$H \times W \times 64$
2D5_2	DeConvGR, filter=3×3, stride=2	2D6_0	$H \times W \times 32$
[2D6_0, 2D2_2]	ConvGR, filter=3×3	2D6_1	$H \times W \times 32$
2D6_1	ConvGR, filter=3×3	2D6_2	$H \times W \times 32$
2D6_2	DeConvGR, filter=3×3, stride=2	2D7_0	$H \times W \times 16$
[2D7_0, 2D1_2]	ConvGR, filter=3×3	2D7_1	$H \times W \times 16$
2D7_1	ConvGR, filter=3×3	2D7_2	$H \times W \times 16$
2D7_2	DeConvGR, filter=3×3, stride=2	2D8_0	$H \times W \times 8$
[2D8_0, 2D0_2]	ConvGR, filter=3×3	2D8_1	$H \times W \times 8$
2D8_1	ConvGR, filter=3×3	2D8_2	$H \times W \times 8$
2D8_2	ConvGR, filter=5×5, stride=2, padding=2	2D9_0	$H \times W \times 16$
2D9_0	ConvGR, filter=3×3	2D9_1	$H \times W \times 16$
2D9_1	ConvGR, filter=3×3	2D9_2	$H \times W \times 16$
2D9_2	ConvGR, filter=5×5, stride=2, padding=2	2D10_0	$H \times W \times 32$
2D10_0	ConvGR, filter=3×3	2D10_1	$H \times W \times 32$
2D10_1	Conv, filter=3×3	$\{\boldsymbol{F}_i\}_{i=0}^{N-1}$	$H \times W \times 32$
3D U-Net			
$\mathcal{C}$	Conv3DGR, filter=3×3	3D0	$D \times H \times W \times 8$
$\mathcal{C}$	Conv3DGR, filter=3×3, stride=2	3D1	$D \times H \times W \times 16$
3D1	Conv3DGR, filter=3×3, stride=2	3D3	$D \times H \times W \times 32$
3D1	Conv3DGR, filter=3×3	3D2	$D \times H \times W \times 16$
3D3	Conv3DGR, filter=3×3	3D4	$D \times H \times W \times 32$
3D3	Conv3DGR, filter=3×3, stride=2	3D5	$D \times H \times W \times 64$
3D5	Conv3DGR, filter=3×3	3D6	$D \times H \times W \times 64$
3D6	DeConv3DGR, filter=3×3, stride=2	3D7	$D \times H \times W \times 32$
3D7+3D4	DeConv3DGR, filter=3×3, stride=2	3D8	$D \times H \times W \times 16$
3D8+3D2	DeConv3DGR, filter=3×3, stride=2	3D9	$D \times H \times W \times 8$
3D9+3D0	Conv3D, filter=1×1	$\mathcal{P}$	$D \times H \times W \times 1$

Table 1. The network details of 2D U-Net and 3D U-Net in VA-MVSNet. We denote Conv, DeConv, Conv3D, DeConv3D as 2D convolutional filter, 2D deconvolutional filter, 3D convolutional filter, 3D deconvolutional filter and use GR to represent the abbreviation of group normalization and the Relu. '+' and '[]' represent the element-wise addition operation and concatenation. N, H, W, D denote input view number, image height, image height and depth hypothesis number.

Fig. 2. Point cloud reconstruction results on *Tanks and Temples* [20] benchmark.