A General Feature Fusion Paradigm for RGB-D Semantic Segmentation[1]

Haoming(Hammond) Liu - Mentor: Prof. Li Guo

Abstract

In semantic segmentation tasks based on RGB-D features, there are two fundamental problems: 1) how to effectively integrate complementary information from depth features and RGB features, and 2) how to enhance high-level semantic information with the fine-grained low-level features. Essentially, both of these problems can be categorized as a challenge of fusing feature maps effectively. This project explores the possibility of merging these two problems and further proposes a general paradigm for fusing feature maps of the same size. The effectiveness of the proposed General Fusion Network has been shown by a solid amount of experiments, and its performance is statistically equivalent to the current state-of-the-art model based on the mIoU measure on the NYUD v2 dataset.

1. Introduction

In the computer vision field, semantic segmentation is a fundamental task that labels each pixel with an object class. Corresponding research has wide applications in many industries, such as human-robot interaction and autonomous vehicle systems. With the prosperity of commercial RGB-D sensors in recent years, we can leverage additional geometric information to solve the above limitations and further improve the performance of semantic segmentation.

Figure 1. (a) RGB image; (b) Depth image; (c) Human-labeled ground-truth (intended output)

However, most existing semantic segmentation models tend to fuse feature maps by simple addition or concatenation, which has not fully exploited the potential of feature fusion. By incorporating a self-designed attention module that can adaptively refine the feature map, this project proposes a simple yet effective paradigm for fusing general feature maps.

[1] More details about this project will be captured by a final paper and submitted to an academic conference.

2. Main Work & Experiment

The main work of this project can be summarized as follows:

(a) This project proposes a novel and lightweight Reduce Pyramid Attention Module, which can adaptively reweight the feature map and highlight important features. Experiment results demonstrate that the RPA Module can significantly improve prediction accuracy after inserting into a baseline model.

(b) This project captures the convention of existing feature fusion methods and introduces a paradigm to generalize the fusion procedures. On top of that, this project proposes a novel Channel-aligned Group-conv operator, which draws the advantages of both addition and concatenation within a fair computational cost.

2.1 Reduce Pyramid Attention Module

The Reduce Pyramid Attention Module is inspired by two outstanding attention modules, SE Module[2] and SPA Module[3]. The SE Module applies global average pooling to all channels and passes the values through a light MLP to generate the weights for channel-wise refinement; whereas the SPA Module enlarges the size of the MLP by constructing a pooling pyramid as input, which undoubtedly secures a better performance yet introduces much more parameters. The proposed RPA Module preserves such merit with a shared FC layer, representing channel-wise features with fewer dimensions and fixing the size of the hidden layer to further reduce the module complexity. The experiment results and module design are as follows:

Attention Module	Accuracy Increment	Parameter Size (c = 64, 128, 256, 512)
SE Module	+1.3%	$c^2/4$
SPA Module	+2.7%	$58c^2$
RPA Module	+2.8%	288c

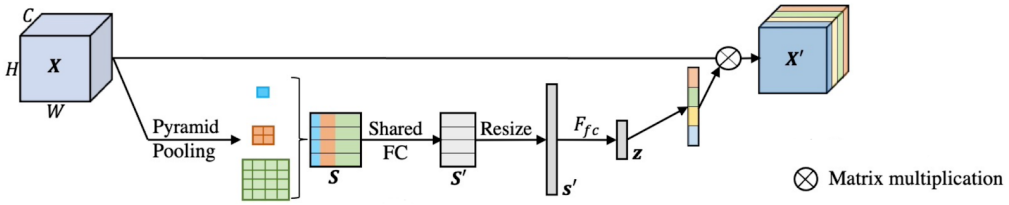


Figure 2. Reduce Pyramid Attention Module

[2] Squeeze-and-Excitation Network (2018): https://arxiv.org/abs/1709.01507
[3] Spatial Pyramid Attention Network (2020): https://ieeexplore.ieee.org/document/9102906

2.2 Channel-aligned Group-conv Operator

Addition and concatenation are two common choices of merging same-size feature maps, but they all have their own disadvantages. Addition is a default choice since it's the lightest operation for merging, yet the channelized weighing scale and correspondence are neither ensured. Lamb-addition and norm-addition balance the scaling issue of different feature maps to some extent, but the reweighting is still from a global perspective. Concatenation followed by convolution is theoretically the best way of merging two feature maps since it applies a linear transformation to the whole feature map. However, this approach converges quite slowly and costs a lot more computational resources. This project proposes a channel-aligned method based on group convolution, which reweighs and merges the feature maps without the loss of channel correspondence. In essence, it applies lamb-addition merge to each channel respectively. The experiment results and module design are as follows:

Merge Operator	Accuracy Increment	Parameters (per op)[4]
Add	+0.00%	0
Norm-Add	+0.34%	8
Lamb-Add	+0.41%	1
Channel-aligned Group-conv	+0.63%	c
Concat-Conv	+0.66%	c^2

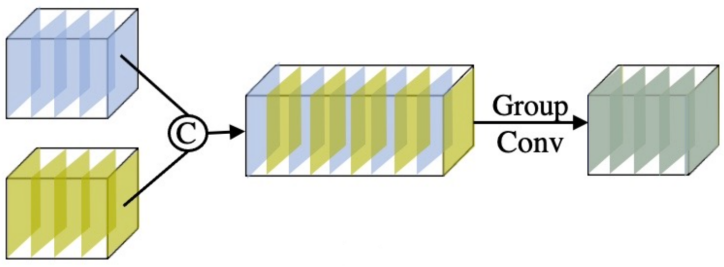


Figure 3. Channel-aligned Group-conv Operator

[4] Parameter *c* here is the number of channels

2.3 RGB-D Semantic Segmentation with General Fusion Module

Guided by the reweigh-and-merge paradigm for feature fusion, this project proposes a general fusion module that can adaptively fuse two same-size feature maps. To start with, the input feature maps reweigh themselves by passing through the RPA Module, then the two refined feature maps can be merged together by a properly chosen merge operator (e.g. add, lamb-add, channel-aligned group-conv, etc.).

The baseline model for RGB-D semantic segmentation originates from ESA Net[5], the current state-of-the-art model on the NYUD v2 dataset. Notably, all the fusion models in the network have been unified to the proposed General Fusion Module. Besides, the network also incorporates multi-scale supervision to speed up training and improve the accuracy of prediction. The model structure and experiment results are as follows:

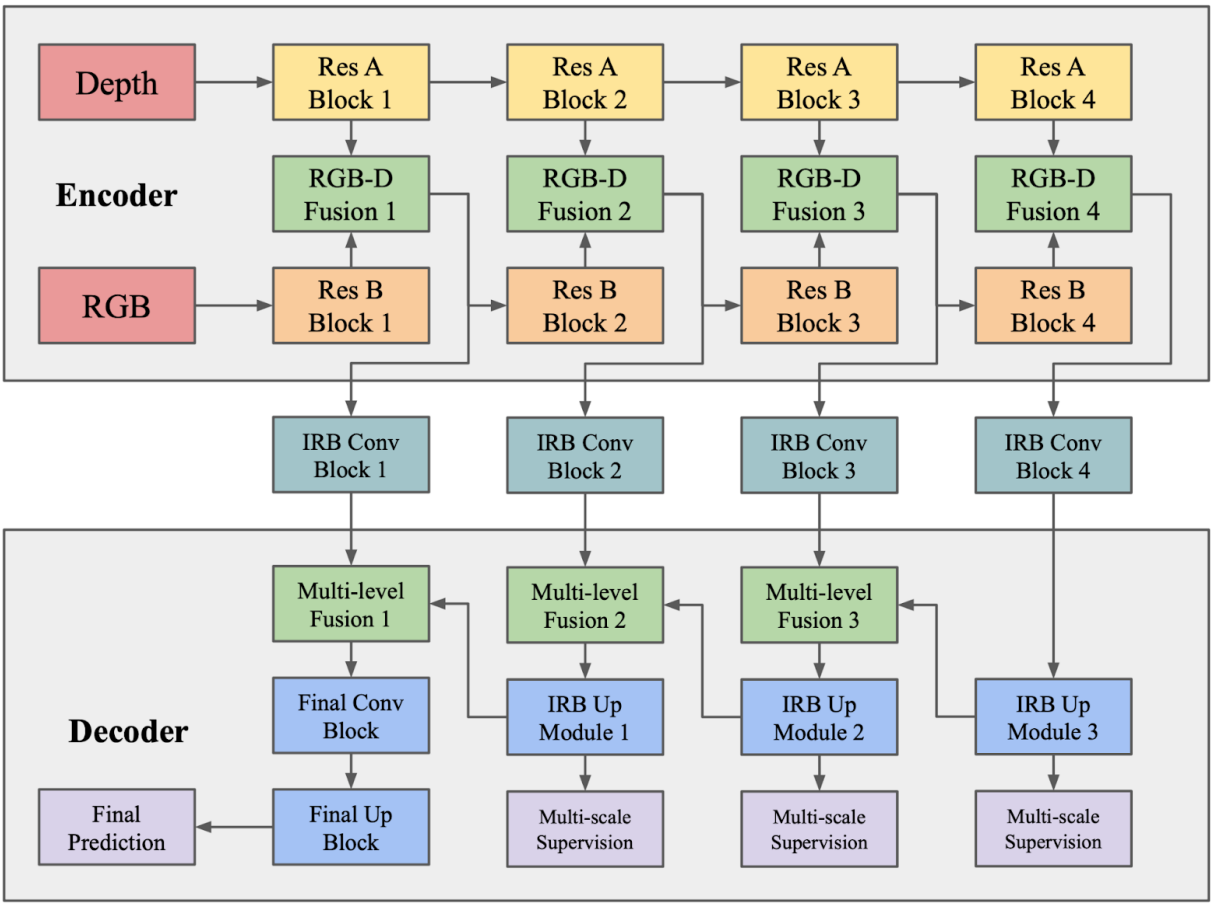


Figure 4. General Fusion Network for RGB-D Semantic Segmentation

[5] Efficient Scene Analysis Network (2021): https://arxiv.org/abs/2011.06961v3

Table 1. Ablation study with ResNet18 backbone on the NYUDv2 dataset

RGB-D Fusion Module (Merge Operator)	Level Fusion Module (Merge Operator)	mIoU (Accuracy Increment*)
None (Add)	None (Add)	46.1% (+0.0%)
None (Add)	RPA Module (Add)	47.1% (+1.0%)
None (Add)	RPA Module (Channel-aligned Group-conv)	47.4% (+1.3%)
RPA Module (Channel-aligned Group-conv)	None (Add)	47.5% (+1.4%)
RPA Module (Add)	None (Channel-aligned Group-conv)	48.2% (+2.1%)
SE Module (Lamb-add)	RPA Module (Channel-aligned Group-conv)	48.6% (+2.5%)

* Note: The baseline model for ablation study uses a more sophisticated structure and applies auxiliary loss, which is different from the simple baseline used for module evaluation in the previous sections. Hence, the increment here is less significant.

Table 2. Comparison with other state-of-the-art methods on the NYUDv2 dataset

Network	Author & Year	Backbone*	mIoU
RDFNet	Park, Hong, and Lee (2017)	2 × R50	47.7%
ESANet	Seichter *et al.* (2021)	2 × R18	48.2%
ACNet	Hu *et al.* (2019)	2 × R50	48.3%
GCNet	This Project (2021)	2 × R18	48.6%
SGNet	Chen *et al.* (2021)	R101	49.0%
Idempotent	Xing *et al.* (2019)	2 × R101	49.9%
SA-Gate	Chen *et al.* (2020)	2 × R50	50.4%
ESANet	Seichter *et al.* (2021)	2 × R50	50.5%

* Note: R18, R50, R101 is the layer-num of the feature extraction ResNet (backbone). In general, dual networks with 50-or-more layer backbones can hardly achieve real-time prediction. Essentially, most of such models are useless for industrial applications.

3. Conclusion

The main innovation of this project is the novel Reduced Pyramid Attention Module and the Channel-aligned Group-conv Operator. Experiment results demonstrate that both of them can significantly improve the prediction accuracy with a fair increment of model complexity. On top of that, this project proposes a light General Fusion Network for RGB-D semantic segmentation, which can fuse multi-modal and multi-level features effectively and identify the object class of pixels accurately. According to the evaluation on the NYUD v2 dataset, the proposed network achieves a state-of-the-art level performance compared with other lightweight models.

Appendix

A. Performance of the General Fusion Network on the NYUD v2 dataset

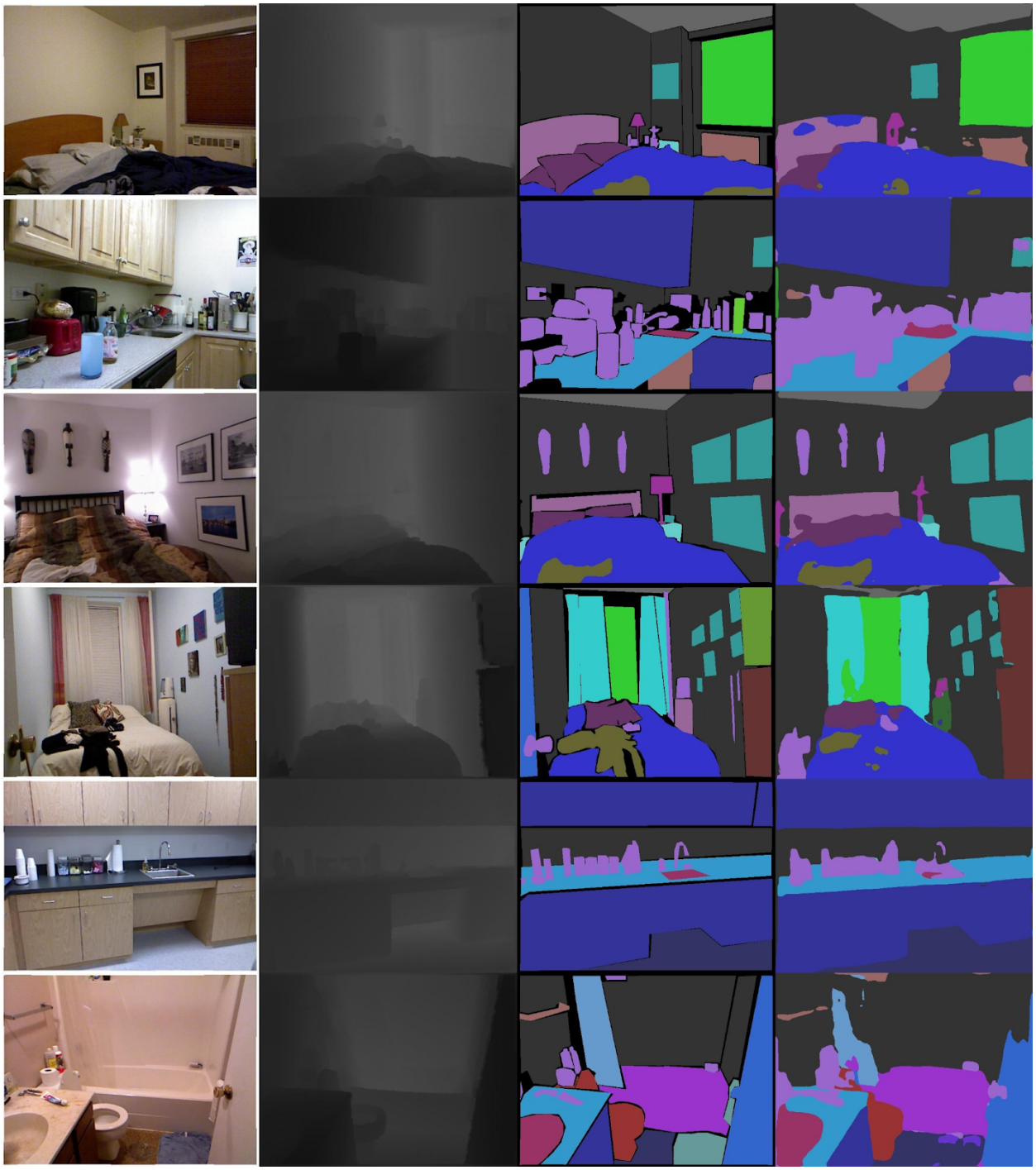
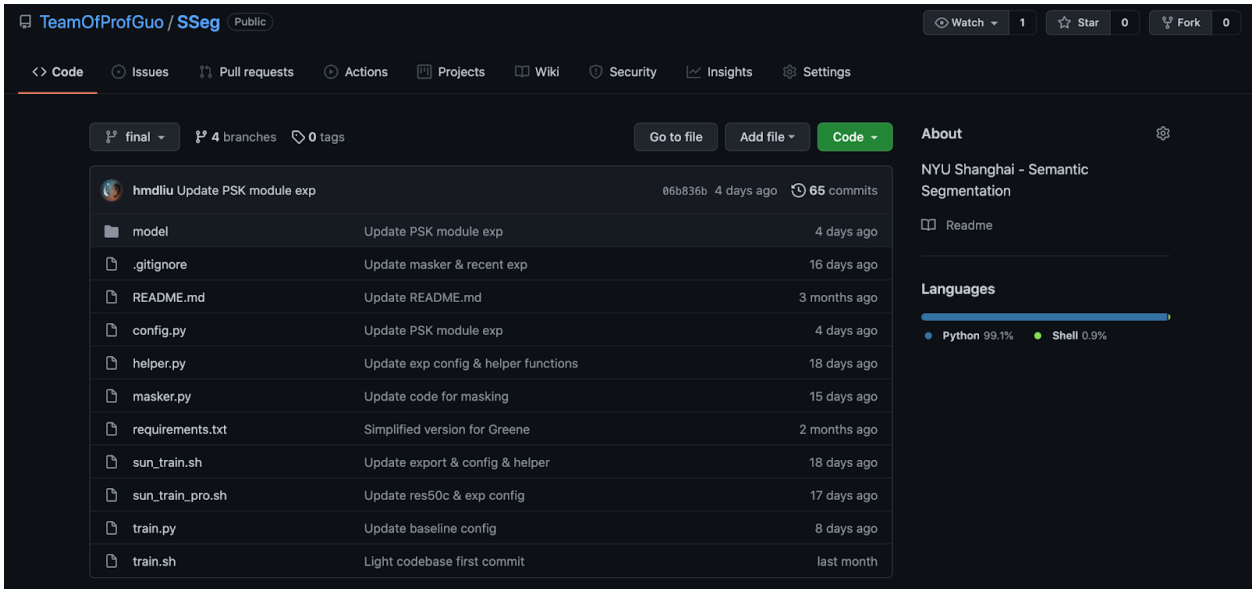
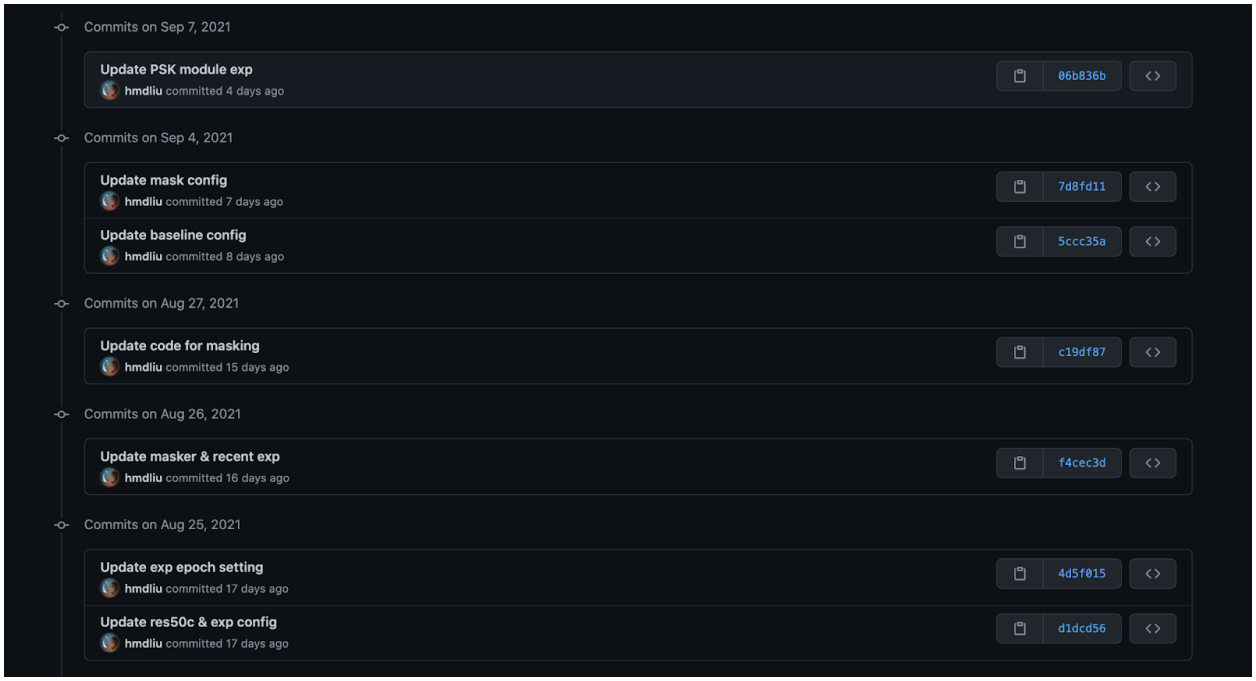


Image Order: RGB input | Depth input | Human-labeled ground-truth | Network prediction

B. [Codebase of the General Fusion Network on Github]

- The source code is open to the public.
- There were 65 code commits throughout the summer.





C. Screenshots of experiments

- The experiments are carried out on the HPC of NYU Shanghai and NYU New York.
- There were more than 1500 experiments for optimizing network baseline and modules.
- On average, each experiment takes about 8 hours to complete.

```
[hl3797@log-2 0820_SSeg]$ squeue -u hl3797
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           9926931      v100     sseg   hl3797  R       0:04      1 gv10
           9926932      v100     sseg   hl3797  R       0:04      1 gv10
           9926930      v100     sseg   hl3797  R       0:07      1 gv10
           9926929      v100     sseg   hl3797  R       0:10      1 gv09
           9926927      v100     sseg   hl3797  R       1:01      1 gv09
           9926926      v100     sseg   hl3797  R       1:04      1 gv08
           9926925      v100     sseg   hl3797  R       1:14      1 gv08
           9926924      v100     sseg   hl3797  R       1:17      1 gv08
           9926922      v100     sseg   hl3797  R       1:38      1 gv08
           9926921      v100     sseg   hl3797  R       1:41      1 gv07
           9926920      v100     sseg   hl3797  R       1:44      1 gv07
           9926919      v100     sseg   hl3797  R       1:48      1 gv07
           9926916      v100     sseg   hl3797  R       2:22      1 gv03
           9926917      v100     sseg   hl3797  R       2:22      1 gv06
           9926915      v100     sseg   hl3797  R       2:25      1 gv06
           9926914      v100     sseg   hl3797  R       2:28      1 gv05
           9926913      v100     sseg   hl3797  R       2:31      1 gv03
           9926911      v100     sseg   hl3797  R       3:00      1 gv02
           9926910      v100     sseg   hl3797  R       3:04      1 gv02
           9926909      v100     sseg   hl3797  R       3:07      1 gv01
           9926908      v100     sseg   hl3797  R       5:54      1 gv01
```

2021-08-11	NYUD v2	GCGF Experiment Greene HPC	b00: aux weight diff	0.4672 / 0.4717	0.7420 / 0.7438	0811a_b00t1.log	600 epochs / 10.20h
				0.4749 / 0.4786	0.7431 / 0.7449	0811a_b00t2.log	600 epochs / 10.22h
				0.4736 / 0.4783	0.7438 / 0.7458	0811a_b00t3.log	600 epochs / 10.35h
				0.4605 / 0.4658	0.7389 / 0.7412	0811a_b00t4.log	600 epochs / 10.32h
			b11: aux weight diff	0.4739 / 0.4783	0.7447 / 0.7452	0811a_b11t1.log	600 epochs / 11.90h
				0.4734 / 0.4788	0.7451 / 0.7468	0811a_b11t2.log	600 epochs / 11.90h
				0.4674 / 0.4720	0.7424 / 0.7444	0811a_b11t3.log	600 epochs / 11.98h
				0.4723 / 0.4781	0.7463 / 0.7468	0811a_b11t4.log	600 epochs / 11.97h
			c14: aux weight diff	0.4721 / 0.4750	0.7449 / 0.7456	0811a_c14t1.log	600 epochs / 12.97h
				0.4826 / 0.4873	0.7479 / 0.7499	0811a_c14t2.log	600 epochs / 12.98h
				0.4699 / 0.4718	0.7433 / 0.7441	0811a_c14t3.log	600 epochs / 12.98h
				0.4715 / 0.4723	0.7460 / 0.7475	0811a_c14t4.log	600 epochs / 12.95h
			c24: aux weight diff	0.4751 / 0.4805	0.7447 / 0.7481	0811a_c24t1.log	600 epochs / 10.67h
				0.4748 / 0.4756	0.7445 / 0.7462	0811a_c24t2.log	600 epochs / 10.67h
				0.4787 / 0.4824	0.7485 / 0.7500	0811a_c24t3.log	600 epochs / 10.57h
				0.4675 / 0.4722	0.7422 / 0.7441	0811a_c24t4.log	600 epochs / 10.60h
			GF + CA6	0.4615 / 0.4668	0.7333 / 0.7352	0811b_a0tt.log	600 epochs / 8.83h
				0.4669 / 0.4717	0.7384 / 0.7401	0811b_a2tt.log	600 epochs / 7.38h
				0.4722 / 0.4778	0.7424 / 0.7449	0811b_b0tt.log	600 epochs / 9.02h
				0.4689 / 0.4737	0.7444 / 0.7444	0811b_b1tt.log	600 epochs / 8.92h
				0.4640 / 0.4688	0.7387 / 0.7406	0811b_c1tt.log	600 epochs / 9.37h
				0.4560 / 0.4603	0.7319 / 0.7350	0811b_c2tt.log	600 epochs / 9.48h