SlimSeg: Slimmable Semantic Segmentation with Boundary Supervision

Danna Xue[1,2], Fei Yang[2], Pei Wang[1], Luis Herranz[2], Jinqiu Sun[1], Yu Zhu[1], Yanning Zhang[1]
[1]Northwestern Polytechnical University, [2]Computer Vision Center, Universitat Autònoma de Barcelona
{danna_xue, wangpei23}@mail.nwpu.edu.cn, {fyang, lherranz}@cvc.uab.es, {sunjinqiu, yuzhu, ynzhang}@nwpu.edu.cn

ACM Multimedia 2022
Lisbon, Portugal | 10-14 October

Motivation

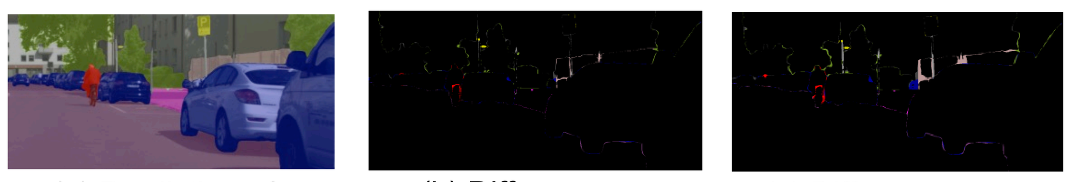
Limitation of Current models:

- Operating at a single accuracy-latency tradeoff
- Relying on well-crafted network architectures
- Conducting slimming only on backbones

→ **dynamic models**

Observation:

- Differences mainly exist along the semantic boundaries

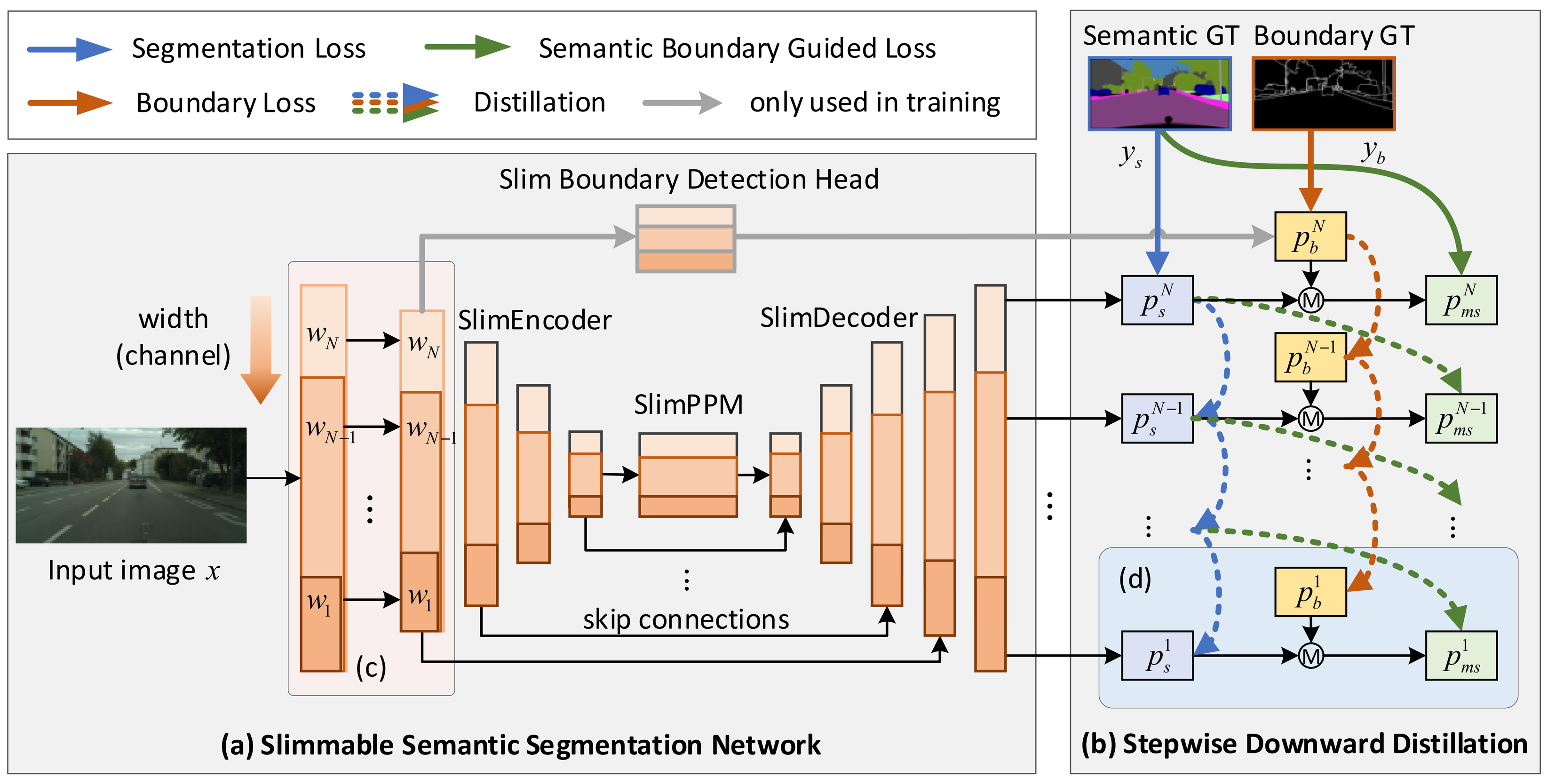


(a) Prediction of model width ×1.0 (b) Difference map ×1.0 vs ×0.75 (c) Difference map ×1.0 vs ×0.25

→ **boundary supervision**

Slimmable Semantic Segmentation Network

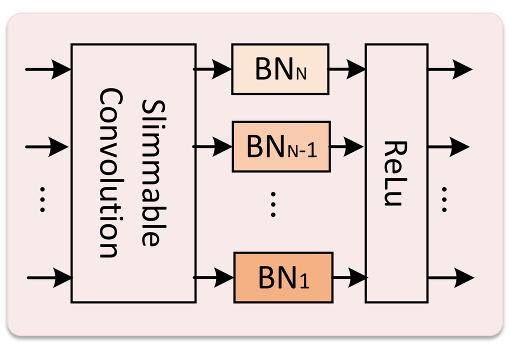
We propose a flexible semantic segmentation framework (SlimSeg), which can adapt its model capacity during inference via the channel slimming mechanism.



Slimmable semantic segmentation framework

Slimmable Network

- The same model runs at different number of **active channels** (widths).
- Independent batch normalization (BN) layers for each widths.



Slimmable Unit

Stepwise Downward Distillation

The class probabilities estimated from the larger subnetwork are used as soft targets for training the next smaller subnetwork.

$$\mathcal{L}_{seg} = \mathcal{L}_{CE}(p_s^N, y_s) + \sum_{n=1}^{N-1} \mathcal{L}_{KD}(p_s^n, p_s^{n+1})$$

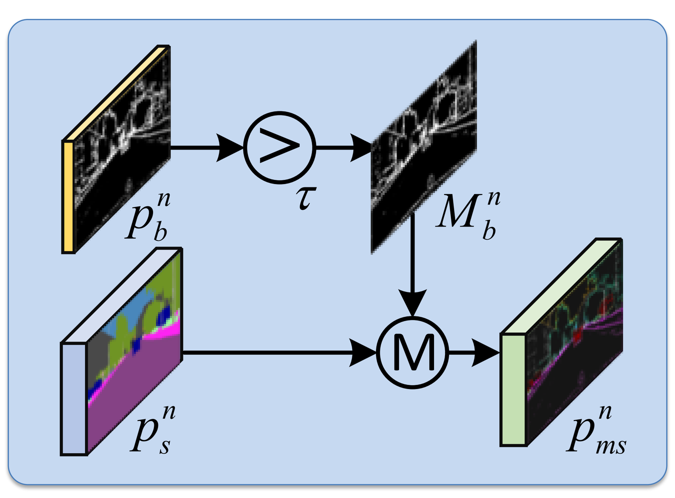
Semantic Boundary Supervision

Auxiliary Boundary Detection Head

$$\mathcal{L}_b = \mathcal{L}_{BCE}(p_b^N, y_b) + \sum_{n=1}^{N-1} \mathcal{L}_{KD}(p_b^n, p_b^{n+1})$$

also training with stepwise downward distillation

Semantic Boundary Guided Loss



Boundary Guided Masking

$$M_b(u,v) = \begin{cases} valid, & p_b(u,v) > \tau \\ invalid, & otherwise \end{cases}$$

$$p_{ms}^n = M_b^n(p_s^n), n \in \{1, 2, \cdots, N\}$$

$$\mathcal{L}_g = \mathcal{L}_{CE}(p_{ms}^N, y_s) + \sum_{n=1}^{N-1} \mathcal{L}_{KD}(p_{ms}^n, p_s^{n+1})$$

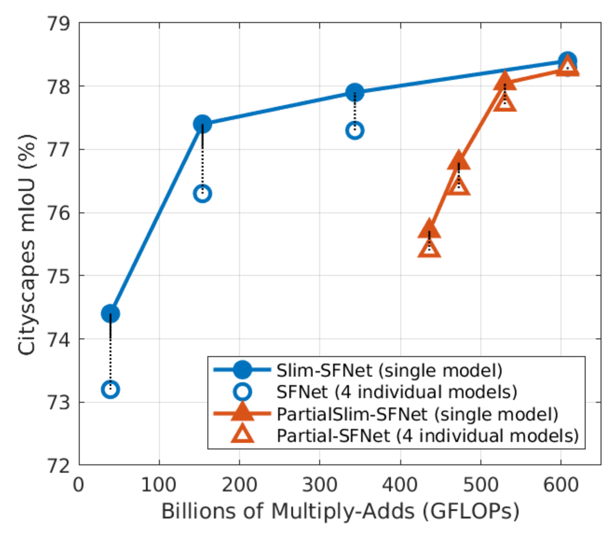
$$\mathcal{L}_{full} = \mathcal{L}_{seg} + \lambda_1 \mathcal{L}_b + \lambda_2 \mathcal{L}_g$$

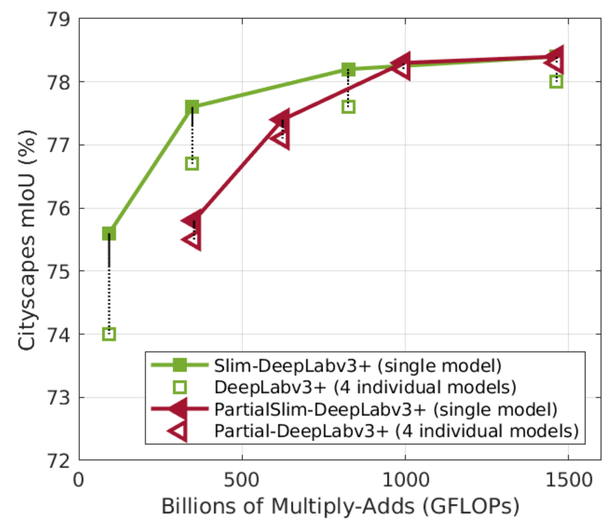
Experiments

Slimmable models vs independent models

- Our slimmable method outperforms the individually trained model especially on large models (ResNet50).
- Globally slimmable models are better than backbone slimmable ones.

Network	Width	Independent mIoU	Independent Param	Slimmable mIoU	Slimmable Param	FLOPs
SFNet ResNet50	×1.0	78.3	31.20	**78.4** (0.1↑)	31.29	607.9
	×0.75	77.3	17.57	**77.9** (0.6↑)		343.4
	×0.5	76.3	7.82	**77.4** (1.1↑)		153.9
	×0.25	73.2	1.97	**74.4** (1.2↑)		39.4
SFNet ResNet18	×1.0	75.0	12.87	**75.6** (0.6↑)	12.89	243.4
	×0.75	74.0	7.24	**74.8** (0.8↑)		137.4
	×0.5	71.4	3.22	**72.5** (1.1↑)		61.5
	×0.25	65.5	0.79	**67.3** (1.8↑)		15.7
SFNet DFNetv2	×1.0	**73.6**	17.88	73.1 (0.5↓)	17.91	80.2
	×0.75	**71.4**	10.06	71.1 (0.3↓)		45.2
	×0.5	**70.0**	4.48	69.8 (0.2↓)		20.2
	×0.25	62.5	1.12	**64.2** (1.7↑)		5.2
SFNet DFNetv1	×1.0	**70.0**	8.42	69.4 (0.6↓)	8.44	32.8
	×0.75	**67.8**	4.74	67.0 (0.8↓)		18.6
	×0.5	65.0	2.11	**65.3** (0.3↑)		8.4
	×0.25	57.8	0.52	**59.8** (2.0↑)		2.2
DeepLabv3+ ResNet50	×1.0	78.0	40.35	**78.4** (0.4↑)	40.44	1463
	×0.75	77.6	22.71	**78.2** (0.6↑)		824.3
	×0.5	76.7	10.11	**77.6** (0.9↑)		347.6
	×0.25	74.0	2.54	**75.6** (1.6↑)		92.9
DeepLabv3+ MobileNetv2	×1.0	66.9	4.53	**67.9** (1.0↑)	4.58	18.5
	×0.75	63.3	2.57	**67.0** (3.7↑)		12.2
	×0.5	58.6	1.16	**64.3** (5.7↑)		5.7
	×0.35	56.1	0.57	**61.1** (5.0↑)		3.3





Stepwise Downward Distillation

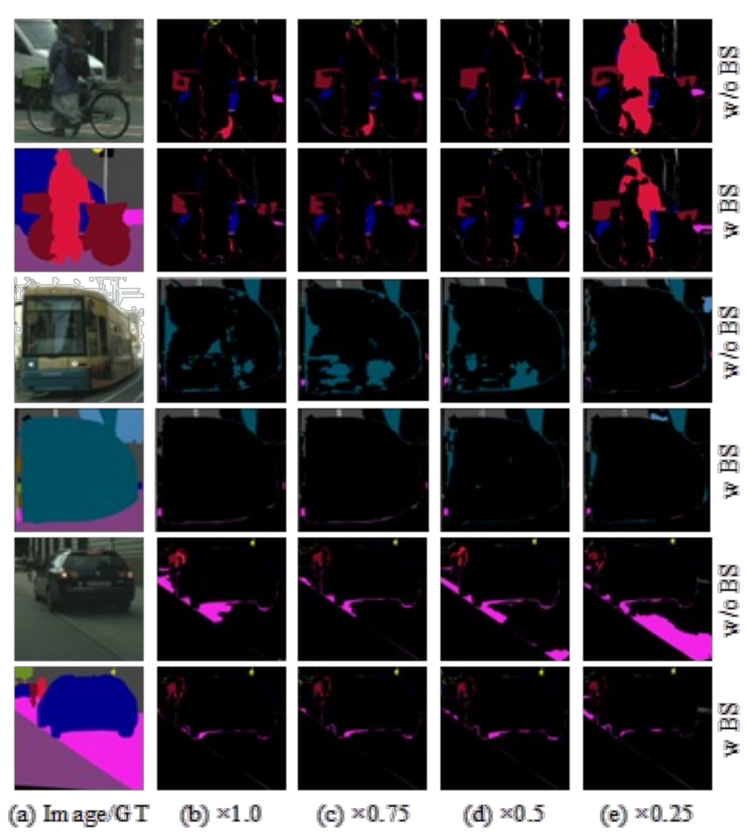
- Using the distillation combined with boundary guided loss function achieve higher accuracy on all the subnetworks.
- Stepwise downward distillation benefits more on smaller submodels.

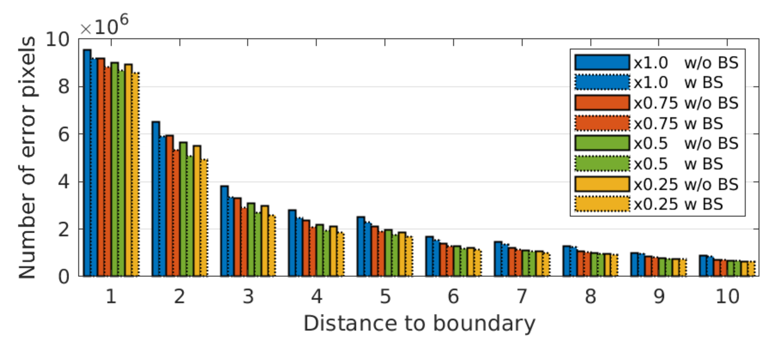
KD	GT $\mathcal{L}_{seg}$	GT $\mathcal{L}_b$	GT $\mathcal{L}_g$	Soft Target $\mathcal{L}_{seg}$	Soft Target $\mathcal{L}_b$	Soft Target $\mathcal{L}_g$	mIoU (%) ×0.25	×0.5	×0.75	×1.0
w/o	✓						71.82	75.97	76.92	77.90
	✓	✓					73.08	76.34	77.12	78.14
	✓		✓				72.49	76.47	77.82	78.35
	✓	✓	✓				73.63	76.92	77.77	78.26
w	✓			✓			71.94	75.86	76.64	77.55
	✓	✓		✓	✓		73.12	76.04	77.21	78.21
	✓		✓	✓		✓	72.94	76.16	77.41	78.37
	✓	✓	✓	✓	✓	✓	**74.40**	**77.37**	**77.87**	**78.43**

KD	Teacher	Loss	mIoU (%) ×0.25	×0.5	×0.75	×1.0
w/o	-	$\mathcal{L}_{CE/BCE}(p^n, y)$	73.63	76.92	77.77	78.26
w	prev	$\mathcal{L}_{KD}(p^n, p^{n+1})$	**74.40**	**77.37**	*77.87*	*78.43*
	largest	$\mathcal{L}_{KD}(p^n, p^N)$	73.64	76.72	77.04	78.38
	mean	$\mathcal{L}_{KD}(p^n, \frac{1}{N-n}\sum_{j=n+1}^{N} p^j)$	73.24	76.25	77.53	77.85
	larger	$\frac{1}{N-n}\sum_{j=n+1}^{N} \mathcal{L}_{KD}(p^n, p^j)$	73.25	75.87	**78.02**	**78.61**

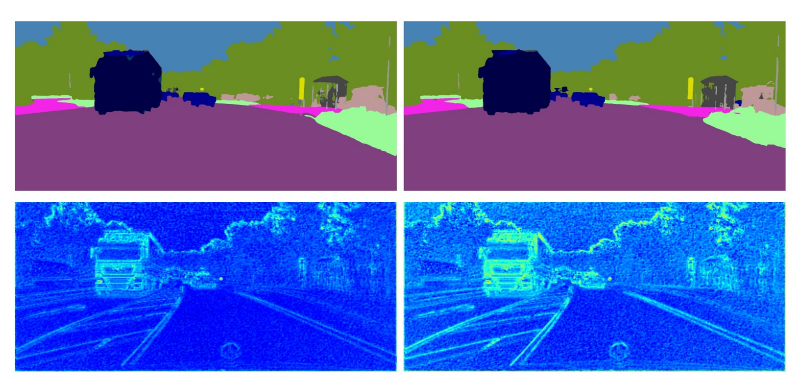
Boundary Supervision

- Better performance on small objects and semantic borders.
- The features in boundary and textured regions are enhanced.





Distribution of error pixels



Low-level features w & w/o BS

Discussion and Limitation

- A general slimmable framework can be directly applied to most mainstream segmentation models.
- Enable adjustable accuracy-efficiency tradeoffs.
- Higher accuracy on smaller submodels without great accuracy drops on large submodels.
- Globally consistent width multipliers are sub-optimal.

Project PID2021-128178OB-I00 funded by