Supplementary Material for the Paper:
MTL-NAS: Task-Agnostic Neural Architecture Search towards General-Purpose Multi-Task Learning

Yuan Gao[1]*, Haoping Bai[2]*†, Zequn Jie[1], Jiayi Ma[3], Kui Jia[4], and Wei Liu[1]

[1] Tencent AI Lab [2] Carnegie Mellon University

[3] Wuhan University [4] South China University of Technology

{ethan.y.gao, bhpfelix, zequn.nus, jyma2010}@gmail.com, kuijia@scut.edu.cn, wl2223@columbia.edu

This supplementary material complements our main paper by giving more details on the following issues:

- Sect. S1 shows the experiments with stronger baselines, *i.e.*, NDDR-CNN (Supernet) and MTL-NAS (Supernet).
- Sect. S2 shows the ablation analysis with different learning rates on the novel inter-task layers.
- Sect. S3 shows the ablation analysis with different combinations on continuous relaxation (*stochastic* and *deterministic*), discretization (*stochastic* and *deterministic*), and entropy minimization.
- Sect. S4 gives illustrations of the learned architectures.

All the additional experiments and ablation analyses in this supplementary material are performed on the NYU v2 [4] dataset for semantic segmentation and surface normal estimation.

S1. Experiments with Stronger Baselines

We perform additional experiments with stronger baselines in this section. The results are shown in Table S1, where NDDR-CNN (Supernet) adds inter-task edges between every pair of nodes *at the same CNN level*, and MTL-NAS (Supernet) uses all the inter-task edges that are included in the search space.

Table S1 shows that our method outperforms both the NDDR-CNN (Supernet), and more interestingly, the MTL-NAS (Supernet) which takes advantage of the whole search space without pruning. Those results illustrate the necessity of performing neural architecture search (NAS) on the supernet, therefore further demonstrating the promising performance of the proposed method.

	Surface Normal Prediction					**Semantic Seg.**	
	Err (↓)		**Within** t°(%) (↑)			**(%)** (↑)	
	Mean	Med.	11.25	22.5	30	mIoU	PAcc
Single-Task	15.6	12.3	46.4	75.5	86.5	33.5	64.1
Multi-Task	15.2	11.7	48.4	76.2	87.0	33.4	64.2
Cross-Stitch [3]	15.2	11.7	48.6	76.0	86.5	34.8	65.0
NDDR [1]	13.9	10.2	53.5	79.5	88.8	36.2	66.4
NDDR (Supernet)	13.8	10.0	54.7	80.2	89.1	36.5	67.2
MTL-NAS (Supernet)	13.3	9.5	56.4	81.4	89.9	36.1	66.9
MTL-NAS	**12.6**	**8.9**	**59.1**	**83.3**	**91.2**	**37.6**	**67.9**

Table S1. Semantic segmentation and surface normal prediction on the NYU v2 dataset using the VGG-16 network. ↑/↓ represents the higher/lower the better. We also include the results of Table 1 in the main text here for more convenient comparison.

* Equal contribution with a random order.

† Work performed at Tencent AI Lab.

S2. Learning Rates for the Novel Inter-Task Architectures

We are interested in the learning rates for the novel architectures and weights because our method inserts novel interlinks into fixed and well-trained network backbone branches.

In Table S2, we investigate the learning rates that are 1-1000 times of the base learning rate. The results show that the result differences among different learning rates are subtle, demonstrating that the proposed method is not sensitive to different learning rates to well train the novel inter-task architectures.

	Surface Normal Prediction					**Semantic Seg.**	
	Err (↓)		**Within t° (%)** (↑)			**(%)** (↑)	
Scale *w.r.t* base LR	Mean	Med.	11.25	22.5	30	mIoU	PAcc
1x	12.6	8.9	59.1	83.3	91.2	**37.6**	**67.9**
10x	**12.4**	**8.8**	**59.8**	**84.0**	**91.7**	37.5	67.8
100x	**12.6**	**8.8**	**59.8**	83.8	91.5	37.1	67.5
1000x	12.6	9.0	58.9	83.3	91.2	37.1	67.5

Table S2. Effects of different learning rates of the new searched layers. We investigate the learning rates that are 1, 10, 100, and 1000 times the learning rate of the backbone networks, respectively.

S3. Different Combinations of *Deterministic* and *Stochastic* Components in the Search Algorithm

We complement our ablation analysis in Table 4 of the main text. Specifically, as shown in Sect. 4.3 (Connections to DARTS and SNAS) of the main text, by unifying the deterministic differentiable architecture search (DARTS) [2] and the stochastic neural architecture search (SNAS) [5] into a more general single-shot gradient-based search algorithm framework, the proposed method is able to extend them by i) imposing entropy minimization to alleviate the objective bias in DARTS and the sampling variance in SNAS, and ii) enabling different combinations of the *stochastic* and *deterministic* components (*i.e.*, the continuous relaxation and the discretization) as a result of i).

The results in Table S3 show that i) imposing minimum entropy regularization significantly improves the performance for all combination cases of the continuous relaxation and the discretization; ii) *under minimum entropy regularization*, different choices of discretization produce negligible performances when the same continuous relaxation method is used. This is because the uncertainty of α is significantly reduced with minimum entropy regularization, *i.e.*, most α's converge to around 0's and 1's, making the sampling procedure of the stochastic discretization behave much more "deterministic". We show the histograms of the converged α under the deterministic or stochastic continuous relaxation, with or without minimum entropy regularization in Fig. S1, which demonstrate that most α's indeed converge to 0's and 1's with minimum entropy regularization[1].

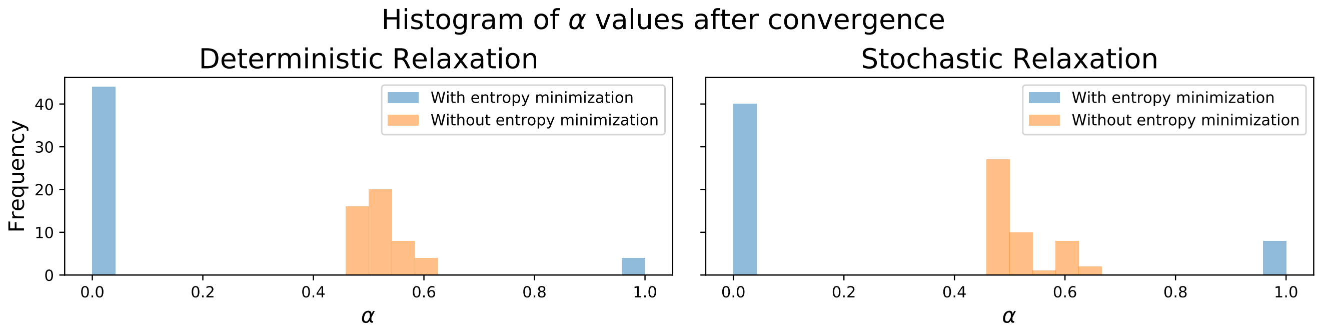

Figure S1. The histogram of converged $alpha$ from *deterministic* (Left) and *stochastic* (Right, the same as Fig. 2 in the main paper) continuous relaxation with and without *minimum entropy regularization*. This shows that the minimum entropy regularization efficiently regularizes the distribution of α's. We draw 25 bins uniformly from 0 to 1, where each bin represents an interval with 0.04.

The phenomena discussed above further demonstrate that the proposed method successfully generalizes and improves the popular DARTS and SNAS methods, and the introduced unified framework is expected to shed light on designing novel single-shot gradient-based search algorithms.

[1] Note that the discretization procedure does not affect the convergence of $alpha$'s, which performs a hard pruning (*i.e.*, deterministic discretization) or a sampling (*i.e.*, stochastic discretization) on the converged α's to deduce the final discrete architectures.

					Surface Normal Prediction					**Semantic Seg.**	
MinEntropy	**Relaxation**		**Discretization**		**Err** (↓)		**Within t° (%)** (↑)			**(%)** (↑)	
	D	S	D	S	Mean	Med.	11.25	22.5	30	mIoU	PAcc
	Random		Search		14.1	10.1	53.9	79.2	88.2	35.3	66.1
	✓		✓		15.9	12.7	45.3	74.0	85.9	19.1	46.1
	✓			✓	-	-	-	-	-	-	-
		✓	✓		14.7	11.1	50.4	77.3	87.7	29.4	60.5
		✓		✓	-	-	-	-	-	-	-
✓	✓		✓		12.7	**8.9**	58.9	83.1	90.9	37.7	**67.9**
✓	✓			✓	12.7	9.0	58.7	82.9	90.9	**37.9**	**67.9**
✓		✓	✓		13.1	9.3	57.0	81.7	90.1	37.3	67.7
✓		✓		✓	**12.6**	**8.9**	**59.1**	**83.3**	**91.2**	37.6	**67.9**

Table S3. Effects of different combinations of continuous relaxation (deterministic and stochastic), discretization (deterministic and stochastic), and entropy minimization. D denotes the *deterministic*, S means the *stochastic*, and MinEntropy represents minimum entropy regularization. Here, Relaxation (D) plus Discretization (D) without MinEntropy is equivalent to DARTS without re-training; Relaxation (S) plus Discretization (S) without MinEntropy is equivalent to SNAS. We did not report the results of methods relying on the Discretization (S) without MinEntropy as those methods produce too large sampling variances. ↑/↓ represents the higher/lower the better.

S4. Illustrations of the Learned Architectures

We illustrate the learned architecture in Fig. S2, which demonstrates that the learned architecture is heterogeneous and asymmetric, therefore being arguably very difficult to be discovered by human experts.

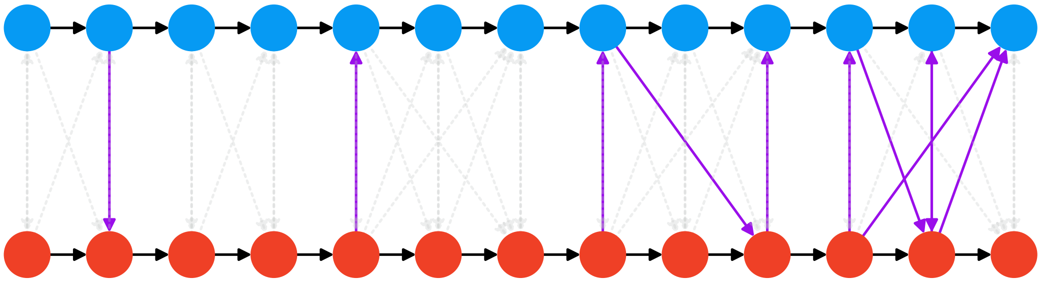

Figure S2. The learned architecture on VGG-16 backbones. Blue/Red nodes and the black solid arrows which link them represent the fixed VGG-16 single task backbones. The gray dash arrows are the candidate searchable edges for inter-task feature fusion (*i.e*., the search space defined by Sect. 5.1 in the main paper), and the purple solid arrows are the learned feature fusion edges.

References

[1] Yuan Gao, Jiayi Ma, Mingbo Zhao, Wei Liu, and Alan L Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *CVPR*, 2019. 1

[2] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *ICLR*, 2019. 2

[3] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *CVPR*, 2016. 1

[4] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 1

[5] Sirui Xie, Hehui Zheng, Chunxiao Liu, and Liang Lin. SNAS: stochastic neural architecture search. In *ICLR*, 2019. 2