MTNAS: Search Multi-Task Networks for Autonomous Driving Supplementary Material

Hao Liu[1], Dong Li[2], JinZhang Peng[2], Qingjie Zhao[1], Lu Tian[2], and Yi Shan[2]

[1] Beijing Institute of Technology, Beijing,CHN {3120181007,zhaoqj}@bit.edu.cn
[2] Xilinx Inc. Beijing,CHN {dongl,jinzhang,lutian,yishan}@xilinx.com

1 Overview

In this supplementary material, we present five sets of algorithm details and additional experimental results.

- We compared DARTS with our method by searching on different datasets.
- We show the performance improvement from the mixed datasets.
- We introduce the details of our final multi-task network.
- We present detailed quantitative evaluations on the mixed-set benchmark.
- We offer more results on different datasets.
- We provide a video demo to show qualitative results of our method.

2 More comparisons with DARTS

We compare the existing NAS method of DARTS with our method by searching on different datasets. Table 1 shows DARTS still obtains inferior performance compared to our MTNAS method in spite of reimplementing it by directly searching on the target datasets. We note that DARTS only optimizes one normal cell and one reduction cell for the entire network while MTNAS optimizes different cells for different branches and backbone. The results show the effectiveness of searching for task-specific branch architectures and task-shared backbone architecture.

Table 1. Comparisons with DARTS by searching on different datasets. We show mAP for detection and mIoU for segmentation on the mixed-set benchmark.

Methods	**mAP (%)**	**mIoU (%)**
DARTS (CIFAR10)	38.6	42.6
DARTS (ImageNet)	35.6	43.0
DARTS (Target)	38.3	44.5
MTNAS (Target)	**43.7**	**46.2**

3 Improvement from Mixed Data

We conduct experiments in Table 2 to show the performance improvement from the mixed datasets. Using mixed training data for detection (Waymo and BDD100K) and segmentation (CityScapes and BDD100K), we can achieve improved performance on either separate or mixed test sets for both tasks.

Table 2. Performance comparisons on the detection and segmentation tasks using mixed or separate training data. W: Waymo. B: BDD100K. C: CityScapes.

Training		Test on Detection			Test on Segmentation		
Detection	Segmentation	W	B	W+B	B	C	B+C
W	B	38.9	-	-	38.4	-	-
W	C	39.2	-	-	-	40.9	-
B	B	-	39.5	-	38.0	-	-
B	C	-	40.4	-	-	41.1	-
W+B	B+C	**39.8**	**42.0**	**40.2**	**42.7**	**44.7**	**44.2**

4 Network Details

We show in Figure 1 our final multi-task network architecture searched on the mixed set. For backbone, we stack 11 normal cells and 3 reduction cells. The normal cells do not change the feature dimension while reduction cells reduce the spatial size of feature maps by half and double the number of channels. For branches, we stack several normal cells and reduction cells by adjusting the kernel size, stride or amount of output channels in the input nodes. For the network searched on the single set, we apply a similar stacking manner but add two more normal cells after each reduction cell in the backbone.

5 Quantitative Evaluations

5.1 Mixed-set result

We also present detailed per-class performance on the mixed-set benchmark in Table 3 and 4. The mixed set includes 4 classes [3] for detection and 16 classes [4] for segmentation. The results show that we achieve consistent improvement for all of classes on both detection and segmentation tasks.

[3] *car*, *pedestrian*, *traffic sign*, *background*

[4] *road*, *sidewalk*, *building*, *wall*, *fence*, *pole*, *traffic light*, *traffic sign*, *vegetation*, *terrain*, *sky*, *person*, *rider*, *car*, *motorcycle*, *bicycle*

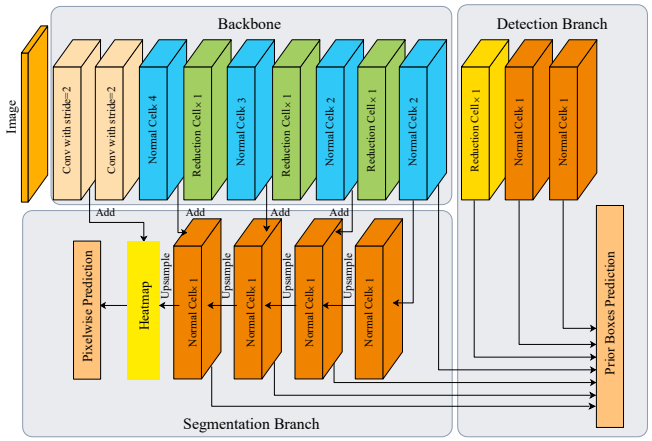

Fig. 1. Illustration of our final multi-task network searched on the mixed set.

Table 3. Performance comparisons of per-class detection accuracy between the multi-task baseline and our MTNAS method on the mixed-set benchmark.

Methods	person	car	traf.sign	mAP
MTL baseline	30.1	59.3	30.1	40.2
MTNAS	35.4	64.3	31.3	43.7

Table 4. Performance comparisons of per-class segmentation accuracy between the multi-task baseline and our MTNAS method on the mixed-set benchmark.

Methods	road	sidewalk	build.	wall	fence	pole	traf.light	traf.sign	
MTL baseline	92.6	59.2	78.8	15.3	22.4	21.2	12.9	30.7	
MTNAS	93.9	61.8	80.5	16.4	24.1	22.7	13.8	32.7	
Methods	vege.	terrian	sky	person	rider	car	motor.	bicycle	mIoU
MTL baseline	79.4	30.0	91.0	46.2	6.2	84.8	5.6	37.0	44.2
MTNAS	80.7	31.7	91.8	49.7	7.4	86.7	7.3	37.3	46.2

5.2 CIFAR10 results

As we can see in Table 5, we search for the network on CIFAR-10 and get similar test error with the CARS method.

5.3 Compare with other NAS methods

We search for the multi-task network on VOC2012 for detection and CityScapes for segmentation with simiar FLOPs to other NAS networks, and results are in Table 6.

Table 5. Performance comparisons of different NAS methods on the CIFAR-10 benchmark.

Method	CARS	Darts	NASNet-A	Random Search	MTNAS
Test Error(%)	2.66	2.76	2.65	3.29	2.66

Table 6. Performance comparisons of different NAS methods on the VOC2012 and CityScapes benchmark.

Method	mAP(%)	mIoU(%)	Search Time(GPU Days)
NAS-FCOS	81.8	-	28
DetNAS	80.1	-	68
Fasterseg	-	71.5	2
Squeezenas-small	-	72.5	14.6
MTNAS	80.6	72.7	20

5.4 Generality

We search for a multi-task network with BDD100K data and evaluate it on the KITTI and CitygScapes benchmarks to assess its generality, and we achieve comparable results with the network which is searched with KITTI and CityScapes data. Results are in Table 7.

Table 7. Performance comparisons of architecture searched with different datasets and evaluate on KITTI&CityScapes.

Search Datasets	mAP(%)	mIoU(%)
BDD100K	68.5	63.5
KITTI&CityScapes	68.8	63.4

6 Qualitative Evaluations

We provide a demo to show the results of our MTNAS method on the BDD100K validation videos. We include different autonomous driving scenes such as highway, downtown, night and rainy day. Our searched model costs 18 ms for each forward propagation on average with an input image of 512×320 on a single NVIDIA P100 GPU.