FAST NEURAL NETWORK ADAPTATION VIA PARAMETER REMAPPING AND ARCHITECTURE SEARCH

报告人：陈宇航

时间：2020年8月13日

Outline

- Motivation
- Method
- Search space
- Experiments

motivation

NAS for detection and segmentation can improve accuracy but one major challenge is:

The ImageNet pre-training of the search space(super net) representation or the searched networks incurs huge computational cost.

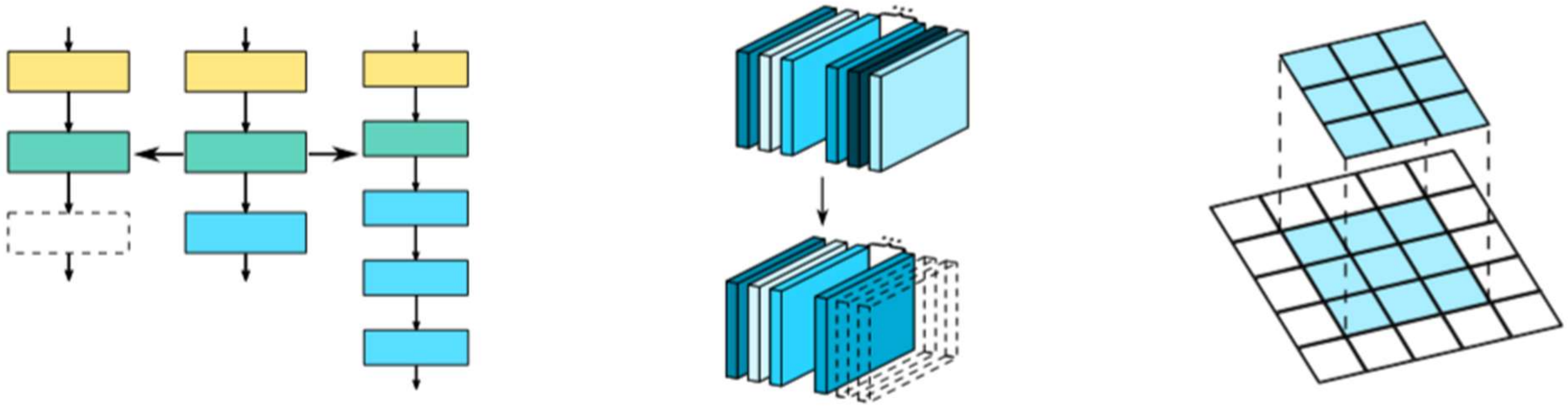
contribution:

propose a Fast Neural Network Adaptation (FNA) method to adapt both architecture and parameters of a seed network to become a network with different depth, width, or kernels via Parameter Remapping.

Outline

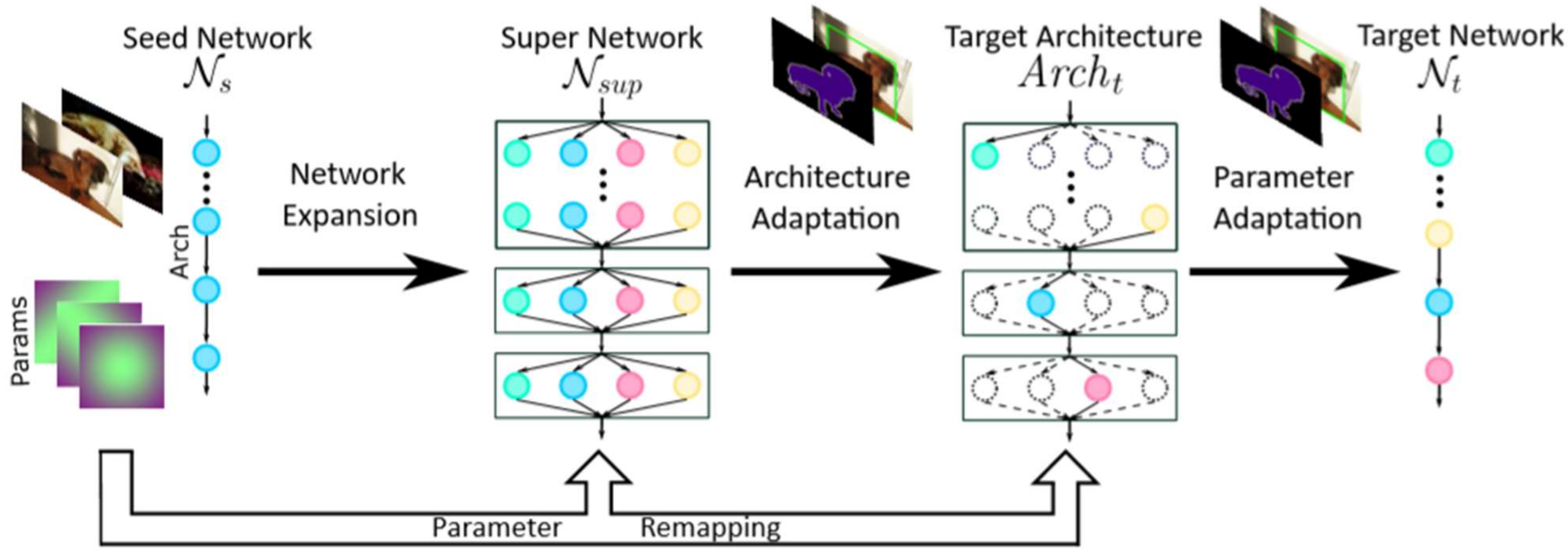
- Motivation
- Method
- Search space
- Experiments

Method



(a) Depth level (b) Width level (c) Kernel level

Method



kernel size settings{3,5,7}
Expansion ratios {1,3,6}
skip_connection

$$\bar{o}^{(i)}(x) = \sum_{o \in O} \frac{exp(\alpha_o^{(i)})}{\sum_{o' \in O} exp(\alpha_{o'}^{(i)})} o(x),$$

Outline

- Motivation
- Method
- Experiments

Experiment

Method		OS	iters	Params	MAdds	mIOU(%)
MobileNetV2 (Sandler et al., 2018)	DeepLabv3	16	100K	2.57M	24.52B	75.5
DPC (Chen et al., 2018a)				2.51M	24.69B	75.4(75.7)
FNA				2.47M	24.17B	**76.6**
Auto-DeepLab-S (Liu et al., 2019a)	DeepLabv3+	8	500K	10.15M	333.25B	75.2
FNA		16	100K	5.71M	210.11B	77.2
FNA		8	100K	5.71M	313.87B	**78.0**

Method	Total Cost	ArchAdapt Cost	ParamAdapt Cost
DPC (Chen et al., 2018a)	62.2K GHs	62.2K GHs	30.0* GHs
Auto-DeepLab-S (Liu et al., 2019a)	244.0 GHs	72.0 GHs	172.0† GHs
FNA	35.8 GHs	1.4 GHs	34.4 GHs

Experiment

Method		Params	MAdds	mAP(%)
ShuffleNetV2-20 (Chen et al., 2019b)	RetinaNet	13.19M	132.76B	32.1
MobileNetV2 (Sandler et al., 2018)		11.49M	133.05B	32.8
DetNAS (Chen et al., 2019b)		13.41M	133.26B	33.3
FNA		11.73M	133.03B	**33.9**
MobileNetV2 (Sandler et al., 2018)	SSDLite	4.3M	0.8B	22.1
Mnasnet-92 (Tan et al., 2018)		5.3M	1.0B	22.9
FNA		4.6M	0.9B	**23.3**

Method	Total Cost	Super Network			Target Network	
		Pre-training	Finetuning	Search	Pre-training	Finetuning
DetNAS (Chen et al., 2019b)	68 GDs	12 GDs	12 GDs	20 GDs	12 GDs	12 GDs
FNA (RetinaNet)	9.2 GDs	-	-	6 GDs	-	3.2 GDs
FNA (SSDLite)	21.6 GDs	-	-	6.6 GDs	-	15 GDs

Experiment

Row Num	Method	MAdds(B)	mIOU(%)
(1)	Remap → ArchAdapt → Remap → ParamAdapt (FNA)	24.17	**76.6**
(2)	RandInit → ArchAdapt → Remap → ParamAdapt	24.29	76.0
(3)	Remap → ArchAdapt → RandInit → ParamAdapt	24.17	73.0
(4)	RandInit → ArchAdapt → RandInit → ParamAdapt	24.29	72.4
(5)	Remap → ArchAdapt → Retrain → ParamAdapt	24.17	76.5

Row Num	Method	MAdds(B)	mAP(%)
(1)	DetNAS (Chen et al., 2019b)	133.26	33.3
(2)	Remap → DiffSearch → Remap → ParamAdapt (FNA)	133.03	**33.9**
(3)	Remap → RandSearch → Remap → ParamAdapt	133.11	33.5
(4)	RandInit → RandSearch → Remap → ParamAdapt	133.08	31.5
(5)	Remap → RandSearch → RandInit → ParamAdapt	133.11	25.3
(6)	RandInit → RandSearch → RandInit → ParamAdapt	133.08	24.9

Experiment

Channel:

$$\boldsymbol{W}_s = (\boldsymbol{W}_s^{(1)} \dots \boldsymbol{W}_s^{(p)})$$

$$\boldsymbol{W}_n = (\boldsymbol{W}_n^{(1)} \dots \boldsymbol{W}_n^{(q)}),$$

$$y_i \leftarrow \gamma \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \beta,$$

$$x_i = (x_i^{(1)} \dots x_i^{(p)})$$

$$|\gamma| = (|\gamma^{(1)}| \dots |\gamma^{(p)}|).$$

Algorithm 1: Weights Remapping Function

```
Input: the seed weights W_s and the new
weights W_n, the reference vector v
// get indices of topk values of the vector
a ← topk-indices(v, k = q)
// sort the indices
sort(a)
for i ∈ 1, 2, ..., q do
    W_n^(i) = W_s^(a[i])
end
Output: W_n with remapped values
```

$$std(\boldsymbol{W}_s^{(1)}) \dots std(\boldsymbol{W}_s^{(p)})$$

$$(||\boldsymbol{W}_s^{(1)}||_1 \dots ||\boldsymbol{W}_s^{(p)}||_1).$$

Experiment

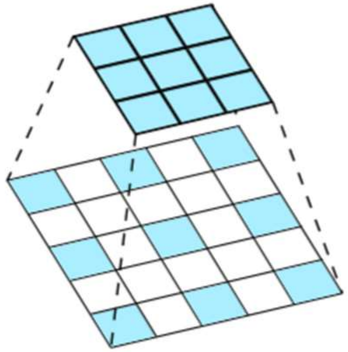


Figure 3: Parameter Remapping on the kernel-level with a dilation setting.

Table 7: Study the methods of Parameter Remapping.

Method	Width-BN	Width-Std	Width-L1	Kernel-Dilate	FNA
mIOU(%)	75.8	75.8	75.3	75.6	**76.6**

The End!