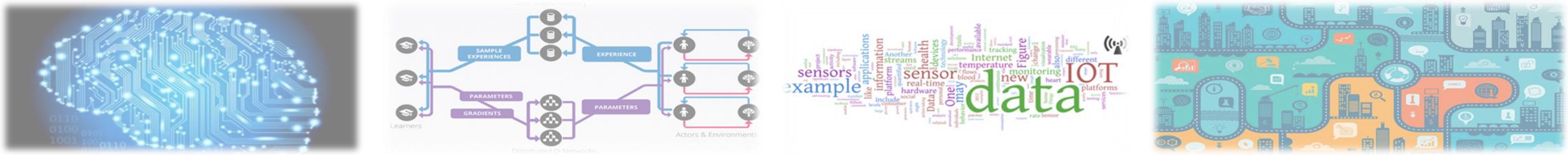

Fast and Practical Neural Architecture Search

AutoML 연구실
석사과정 문아성

21.04.28

Fast and Practical Neural Architecture Search

- FPNAS: Fast and Practical Neural Architecture Search

Cui, Jiequan, et al. "Fast and practical neural architecture search."

Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

- Mobile Efficient NAS

  - Fast Search

  - Mobile inverted bottleneck convolution

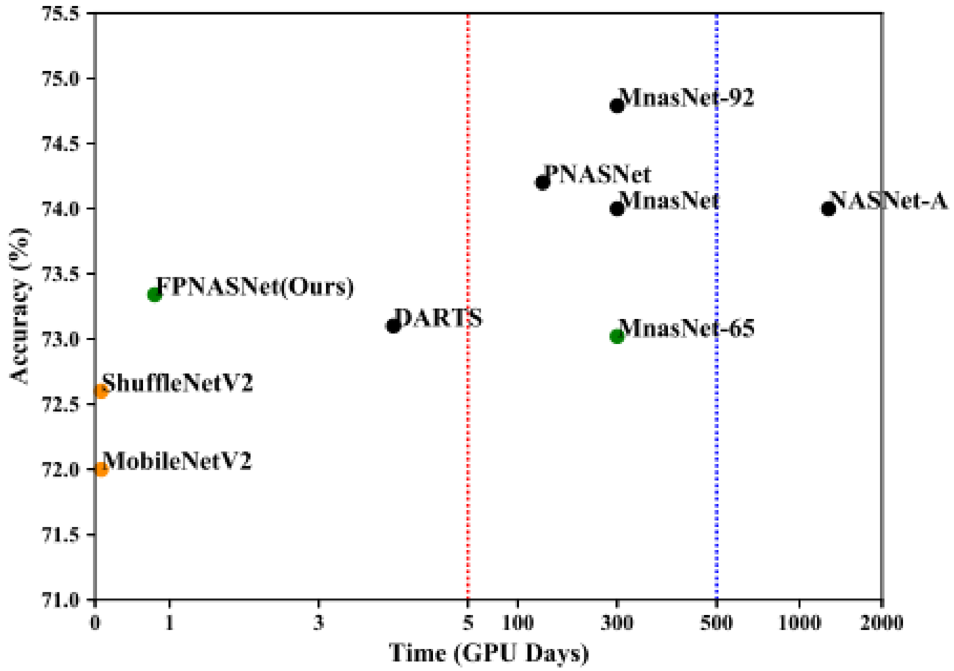

Figure 1. Search time and Top-1 accuracy of different NAS frameworks on ImageNet validation set. Green dots represent networks with less than 300M FLOPs while black ones are those with 300M+ FlOPs. Orange dots are top handcrafted mobile-friendly networks. Our proposed FPNAS is able to find useful networks (FPNASNet) within one GPU day. We note that ENAS [12] is not included in this figure since the performance is not reported on ImageNet.

Fast and Practical Neural Architecture Search

Mobile-efficient Search Space

- Block Architecture
  - $Block_{1:N} = \{Block_1, Block_2, \dots, Block_N\}$
  - $Block$: Directed acyclic graph (DAG)

$$Block = (V, E)$$

- Vertices
  - Element-wise addition
  - Split operation
  - Concat operation
  - Identity mapping
- Edges
  - Mobile inverted bottlenect convolution
  - 3x3 or 5x5 convolution
  - Identity mapping

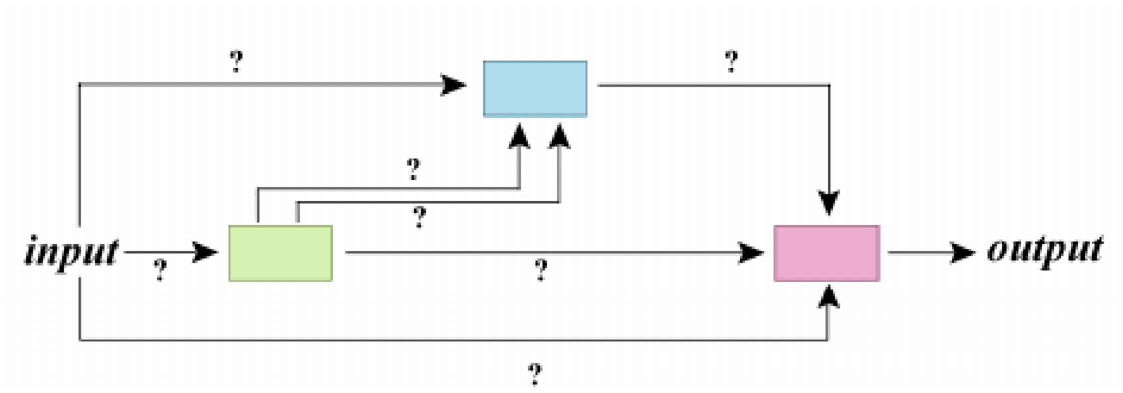

Figure 2. Overview of every block's architecture. Every box represents a vertex and there are at most three vertices, each arrowed line represents an edge. There are two edges from the green vertex to the blue one because the green vertex may be a split operation. Initially, operations on edges and vertices are unknown. We put questions marks for indication.

Fast Search Method

- b: FLOPs

$$\max_{Block_{1:N}} \quad R(Block_{1:N}) \\ s.t. \quad f(Block_{1:N}) < b. \qquad (1)$$

$$R(Block_{1:N}) = \begin{cases} Acc, & f(Block_{1:N}) < b \\ -1, & f(Block_{1:N}) \geq b \end{cases} \qquad (2)$$

$$\max_{x \in X, y \in Y} \quad F(x, y) \\ s.t. \quad x = \arg\max_{x \in X} G(x, y). \qquad (3)$$

$$\max_{Block_{1:N/2}, Block_{N/2+1:N}} \quad R(Block_{1:N}) \\ s.t. \quad Block_{1:N/2} = \arg \max_{Block_{1:N/2}} R(Block_{1:N}) \qquad (4)$$

Experiment Results

Dataset: CIFAR- 10

Model	Params	Error (%)	Inference Speed (Images/s)
ShuffleNet V2 (1.5×) [19]*	2.47M	5.83	8,000
MobileNet V2 (1×) [26]*	2.20M	4.13	16,000
FPNASNet (Ours)	1.68M	3.99	13,500
Hier-EA [16]	15.70M	3.75	-
PNASNet-5 [15]	3.20M	3.41	-
AmoebaNet-A [21]	3.20M	3.34	-
DARTS (first order) [17] + CutOut	3.30M	3.00	-
ENAS [20] + CutOut	4.60M	2.89	900
DARTS (second order) + CutOut	3.30M	2.76	-
NASNET-A [40] + CutOut	3.30M	2.65	-
FPNASNet (2×) + CutOut (Ours)	5.76M	3.01	12,800

Table 1. Results of our FPNASNet on CIFAR-10. They are compared to MobileNet V2, ShuffleNet V2 and other NAS methods. For better performance, the first three layers with stride 2 in MobileNet V2 are changed to 1 and the first two layers with stride 2 in ShuffleNet V2 are set to 1. * denotes our implementation. 'FPNASNet (2×)' means we double all channels. The inference time is tested with PyTorch for MobileNet V2, ShuffleNet V2 and FPNASNet. The inference speed is tested with Tensorflow for ENAS.

Experiment Results

Dataset: ImageNet

Model	Type	GPU Hours	DataSet for NAS	Parameters	Mult-Adds	Top-1 (%)
SqueezeNext [6]	manual	-	-	3.20M	708M	67.50
MobileNet V1 [9]	manual	-	-	4.20M	575M	70.60
CondenseNet (G=C=8) [10]	manual	-	-	4.80M	529M	73.80
MobileNet V2 (1×) [26]	manual	-	-	3.47M	300M	72.00
ShuffleNet V2 (1.5×) [19]	manual	-	-	3.50M	299M	72.60
ShuffleNet (1.5×) [34]	manual	-	-	3.40M	292M	71.50
CondenseNet (G=C=4) [10]	manual	-	-	2.90M	274M	71.00
MobileNet V2 (0.75×) [26]	manual	-	-	2.61M	209M	69.80
MobileNet V1 (0.5×) [9]	manual	-	-	1.30M	149M	63.70
ShuffleNet V2 (1×) [19]	manual	-	-	2.30M	146M	69.40
NASNet-A [40]	auto	32400	CIFAR-10	5.30M	564M	74.00
MnasNet-92 [29]	auto	7000*	ImageNet	4.40M	388M	74.79
MnasNet [29]	auto	7000*	ImageNet	4.20M	317M	74.00
MnasNet-65 [29]	auto	7000*	ImageNet	3.60M	270M	73.02
PNASNet [15]	auto	3600	CIFAR-10	5.10M	588M	74.20
DARTS [17]	auto	96	CIFAR-10	4.90M	595M	73.10
FPNASNet-C (Ours)	auto	**20**	CIFAR-10	1.91M	149M	69.91
FPNASNet-B (Ours)	auto	**20**	CIFAR-10	3.07M	216M	70.67
FPNASNet-A (Ours)	auto	**20**	CIFAR-10	2.95M	245M	72.01
FPNASNet (Ours)	auto	**20**	CIFAR-10	3.41M	300M	73.34

Table 2. Results of image classification on ImageNet. We compare our FPNASNet models with both handcrafted mobile models and other automated approaches. FPNASNet, FPNASNet-A, FPNASNet-B and FPNASNet-C are the searched models (for comparison) with different FLOPs and parameters. #Parameters: number of trainable parameters; #Mult-Adds: number of multiplication-add operations per image; Top-1 Acc: top-1 accuracy on ImageNet validation set. * is our estimation according to the number of models reported in [29] (with about 8K different models).

Fast and Practical Neural Architecture Search

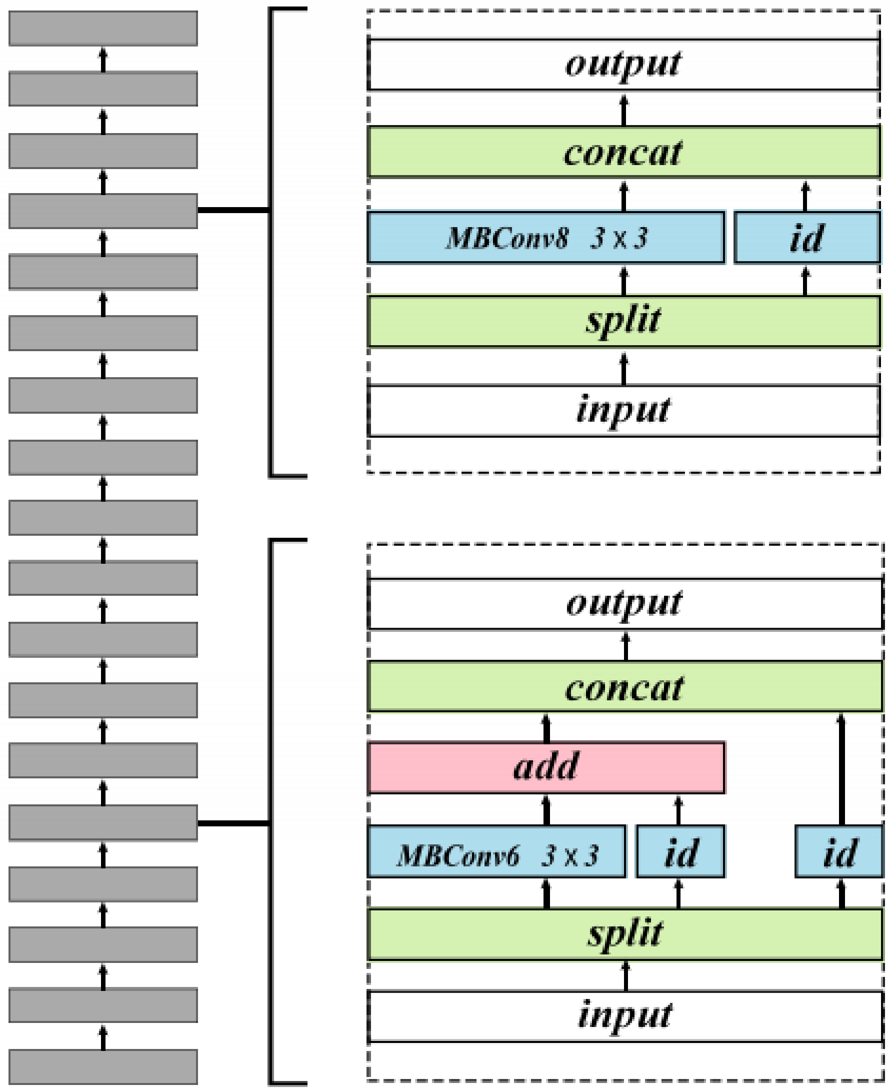

Figure 5. More details of our searched network structure. Blocks at different positions are with their special topological architectures. $MBConv6$ and $MBConv8$ are mobile inverted bottleneck convolution with expansion ratios 6 and 8 respectively.