Network Pruning via Transformable Architecture Search

Xuanyi Dong[1,2], Yi Yang[1]

[1]ReLER, University of Technology Sydney

[2]Baidu Research

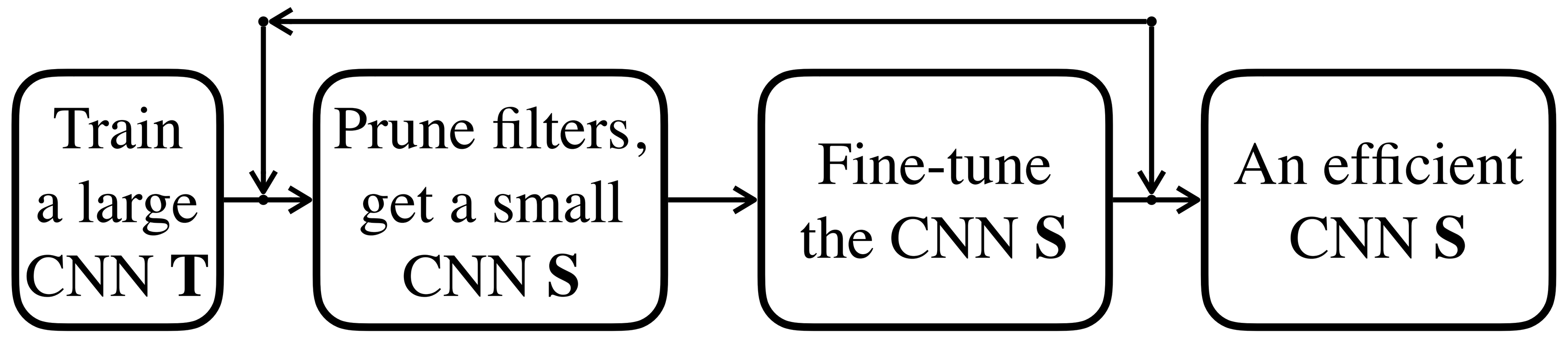
INTRODUCTION

Transformable Architecture Search (TAS): search for the best size of a network, i.e., the width and depth.

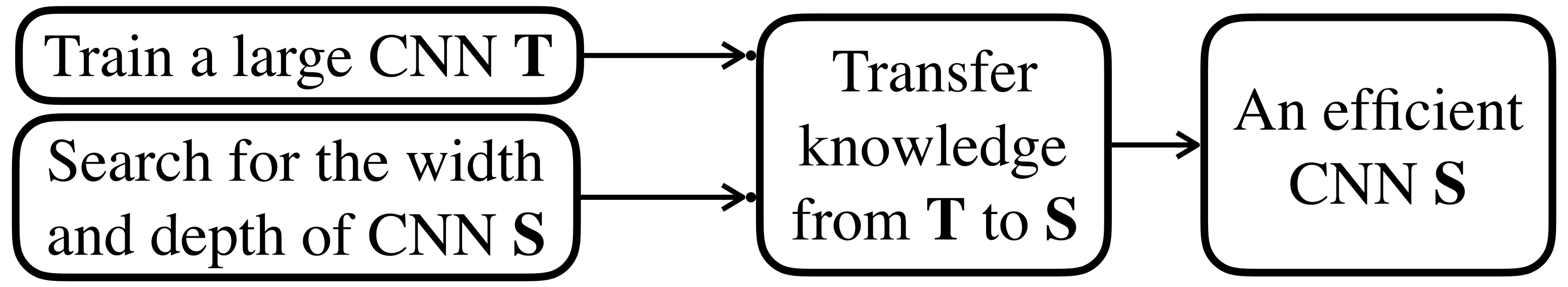
vs.

Traditional Neural Architecture Search (NAS): search for the topology structure of a network.

We proposed a new paradigm for network pruning:
Train a CNN -> Apply TAS for the CNN -> Transfer params.



(a) The Traditional Pruning Paradigm



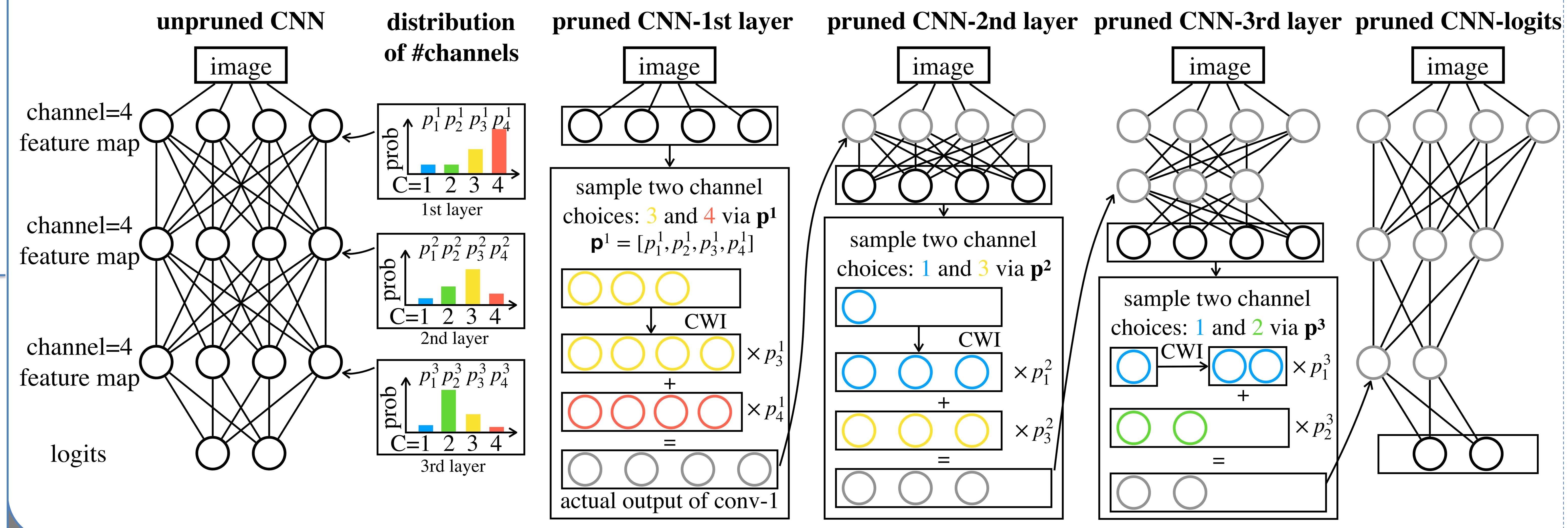
(b) The Proposed Pruning Paradigm

Contribution:
(1) A new pruning paradigm with SOTA performance.
(2) A differentiable searching method for the network shape.

SCAN ME! search codes

video demo for intermediate results

MAIN IDEA OF TAS



Search for the width of a three-layer CNN

Each convolutional layer is equipped with a learnable distribution for the size of the channels in this layer, indicated by p^i on the left side. The feature map for every layer is built sequentially by the layers, as shown on the right side. For a specific layer, K (2 in this example) feature maps of different sizes are sampled according to corresponding distribution and combined by channel-wise interpolation (CWI) and weighted sum. This aggregated feature map is fed as input to the next layer.

OBJECTIVE

$$\mathcal{L} = -\log\left(\frac{\exp(\mathbf{z}_y)}{\sum_{j=1}^{|\mathbf{z}|}\exp(\mathbf{z}_j)}\right) + \lambda_{cost}\mathcal{L}_{cost}$$

$$\mathcal{L}_{cost} = \begin{cases} \log(\mathbb{E}_{cost}(\mathbb{A})) & F_{cost}(\mathbb{A}) > (1+t)R \\ 0 & otherwise \\ -\log(\mathbb{E}_{cost}(\mathbb{A})) & F_{cost}(\mathbb{A}) < (1+t)R \end{cases}$$

$\mathbb{A}$ is the set of parameters modeling the net config
R is the target computational cost, e.g., 300M FLOPs
$\mathbb{E}_{cost}(\mathbb{A})$ is the expectation of costs based on $\mathbb{A}$.
$F_{cost}(\mathbb{A})$ is the actual cost of the searched architecture

IMPORTANCE OF TAS and KD

	FLOPs	accuracy
Pre-defined	41.1 MB	68.18 %
Pre-defined w/ Init	41.1 MB	69.34 %
Pre-defined w/ KD	41.1 MB	71.40 %
Random Search	42.9 MB	68.57 %
Random Search w/ Init	42.9 MB	69.14 %
Random Search w/ KD	42.9 MB	71.71 %
TAS†	42.5 MB	68.95 %
TAS† w/ Init	42.5 MB	69.70 %
TAS† w/ KD (TAS)	42.5 MB	**72.41 %**

Prune 40% FLOPs of ResNet-32 on CIFAR-100

Pre-defined
each layer prune 77% channels
Random Search
pickup the best from 10 random configurations
TAS
automatically search for the best configuration

Init
use pre-trained weights
KD
use knowledge distillation