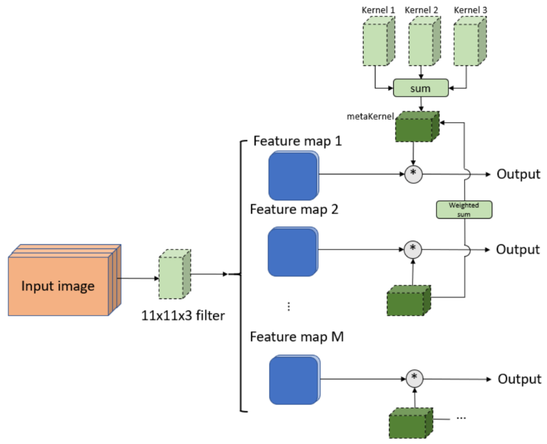
E-DNAS: Differentiable Neural Architecture Search for Embedded Systems

Javier García López[a,b], Antonio Agudo[b] and Francesc Moreno-Noguer[b]
[a]FICOSA ADAS SLU, Barcelona, Spain
[b]Institut de Robòtica i Informàtica Industrial, CSIC-UPC, 08028, Barcelona, Spain

Abstract

In this work we introduce **E-DNAS, a differentiable architecture search method**, which improves the efficiency of NAS methods in designing light-weight networks for the task of image classification.
E-DNAS computes, in a differentiable manner, the optimal size of a number of meta-kernels that capture patterns of the input data at different resolutions. We also leverage on the additive property of convolutions operations to merge several kernels with different sizes into a single one, reducing thus the number of operations. We report results in terms of the SoC (System on Chips) metric, typically used in the Texas Instruments TDA2x families for autonomous driving applications proving good results in terms of accuracy and search time

Method



In this work we propose a methodology for automatic neural architecture design to be executed on embedded platforms. We present a dual step pipeline for the automatic finding of the proper NN to run on a particular platform attending to direct metrics like latency:

- High resolution feature extraction through depthwise convolution using big dimension convolutional kernels.

- Pairwise neural architecture cross-search for the calculated feature maps on previous step.

Experiments and results

Model	Search Method	Search Space	Search Dataset	# Params(M)	FLOPs(M)	acc(%)
MNetV2 [28]	manual	-	-	3.4	300	72.0
CondenseNet(G=C=8) [29]	manual	-	-	4.8	529	73.8
EfficientNet-B0 [30]	manual	-	-	5.3	390	76.3
NASNet-A [5]	RL	cell	CIFAR-10	5.3	564	74.0
PNASNet [31]	SMBO	cell	CIFAR-10	5.1	588	74.2
DARTS [9]	gradient	cell	CIFAR-10	4.7	574	73.3
PDARTS [32]	gradient	cell	CIFAR-10	4.9	557	75.6
GDAS [21]	gradient	cell	CIFAR-10	4.4	497	72.5
MnasNet [17]	RL	stage-wise	ImageNet	3.9	312	75.2
Single-Path NAS [33]	gradient	layer-wise	ImageNet	4.3	365	75.0
ProxylessNAS-R [24]	RL	layer-wise	ImageNet	4.1	320	74.6
ProxylessNAS-G [24]	gradient	layer-wise	ImageNet	-	-	74.2
FBNet [20]	gradient	layer-wise	ImageNet	5.5	375	74.9
MNetV3 Large [34]	RL	layer-wise	ImageNet	5.4	219	75.2
MNetV3 Small [34]	RL	layer-wise	ImageNet	2.9	66	67.4
MixNet [14]	RL	kernel-wise	ImageNet	5.0	360	77.0
MetaKernels [10]	gradient	kernel-wise	ImageNet	7.2	357	77.0
Ours	**gradient**	**parallel kernel-wise**	**ImageNet**	**5.9**	**365**	**76.9**

Results on ImageNet classification Benchmark

One of the main contributions is the proposed **circular feedback on each iteration** to speed up the process by updating the target weights and network parameters iteratively
We test out method with commercial hardware used in the autonomous driving industry obtaining good results in terms of Accuracy and search time

Results

Model	# Params (M)	MACs (M)	Time (ms)
MNet [2]	4.2	569	75
NasNet-A [5]	5.3	564	183
Ours	**5.9**	**535**	**38**

We propose a two-step pipeline that learns different meta-kernel sizes, able to treat different resolution patterns to create automatic neural networks that can classify images. Our method can automatically design light and fast neural networks that can fit on a target embedded platform, such as, the one proposed in this work: Texas Instruments TDA2x family.