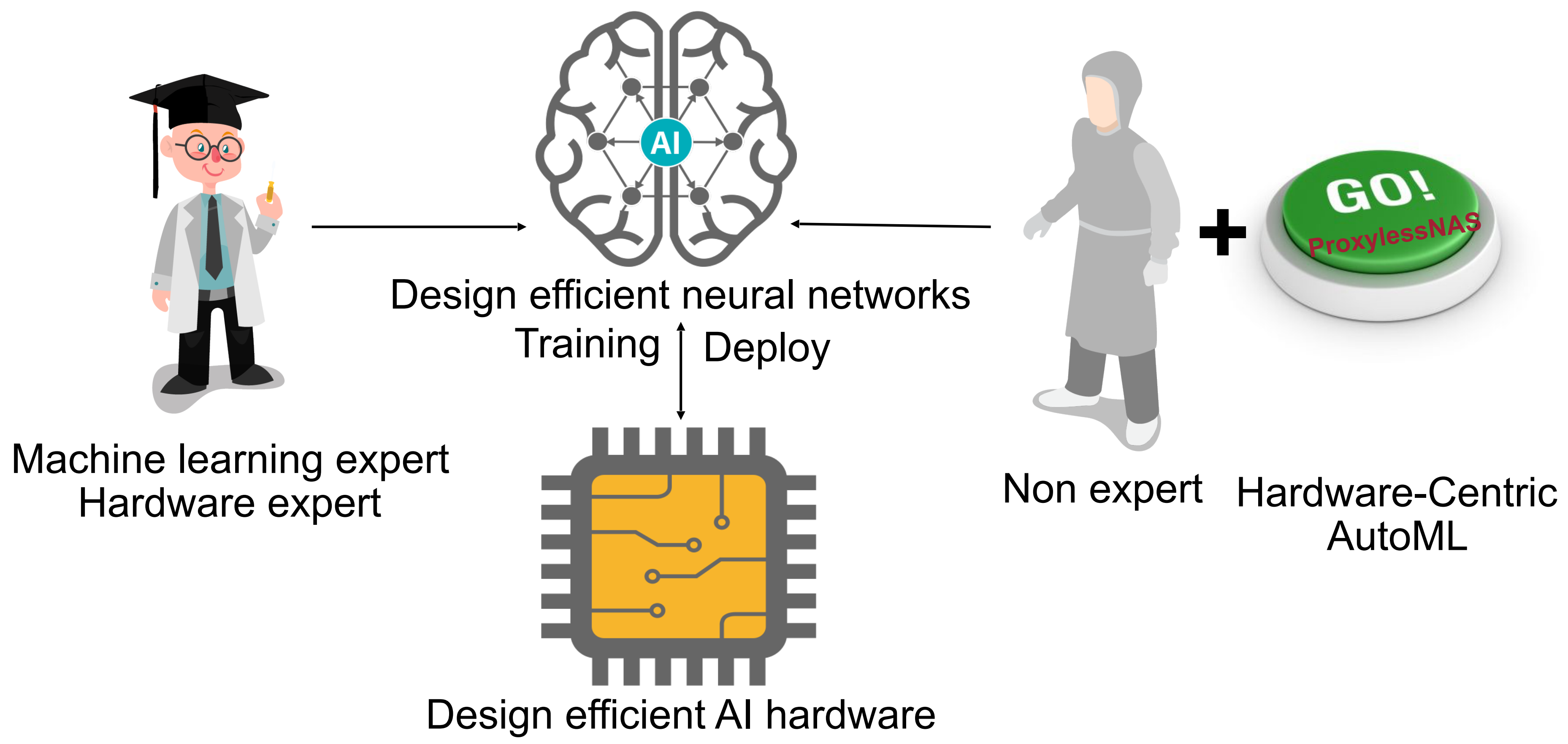
HAN LAB Hardware, AI and Neural-nets

ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware

Han Cai, Ligeng Zhu, Song Han
Massachusetts Institute of Technology

GitHub

Design Automation for Hardware Efficient Nets

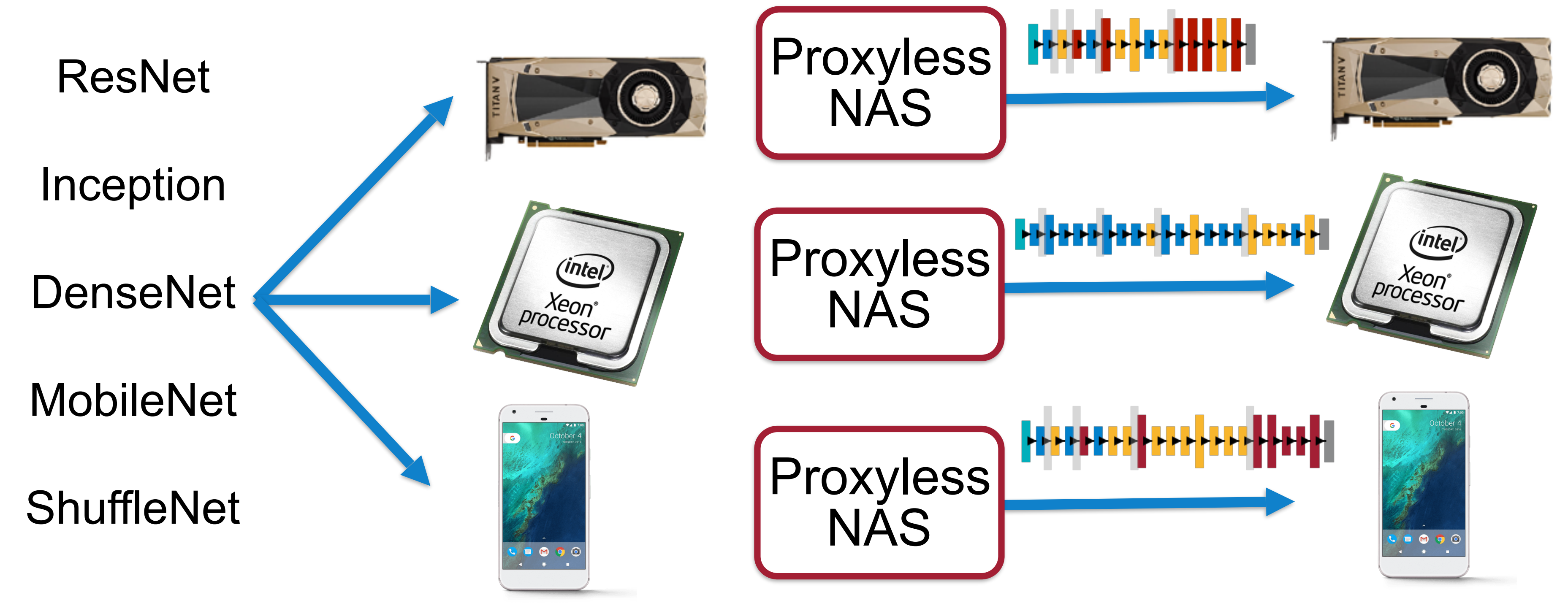


Hardware-Centric AutoML allows non-experts to efficiently design neural network architectures with a push-button solution that runs fast on a specific hardware.

From General Design to Specialized CNN

Previous Paradigm:
One CNN for all Platforms

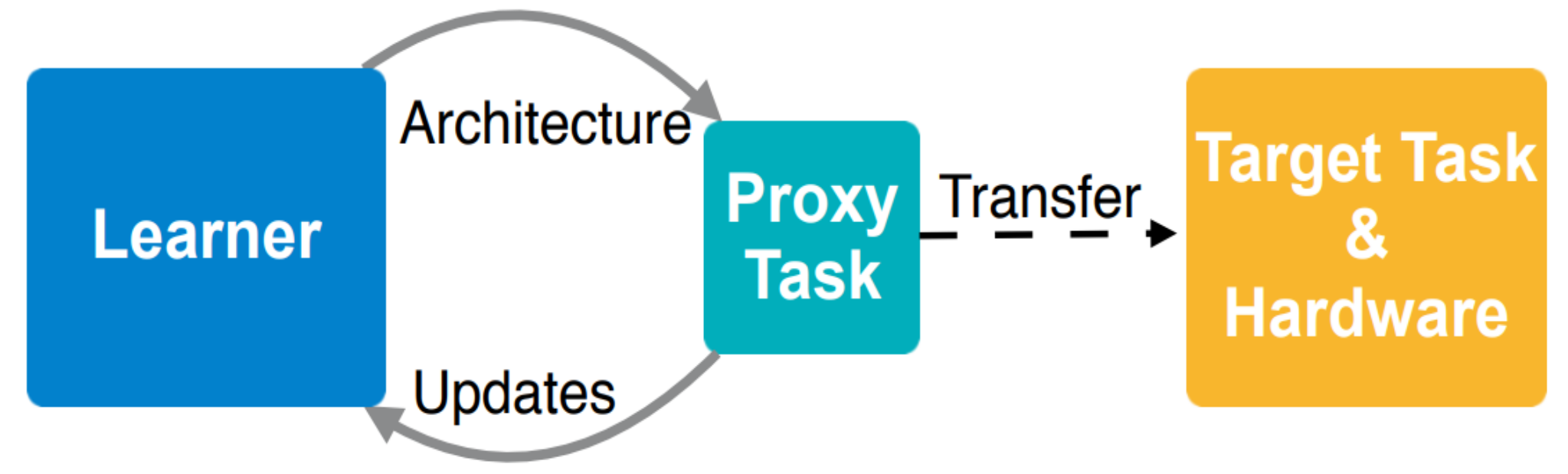
Our Work:
customize CNN for each platform



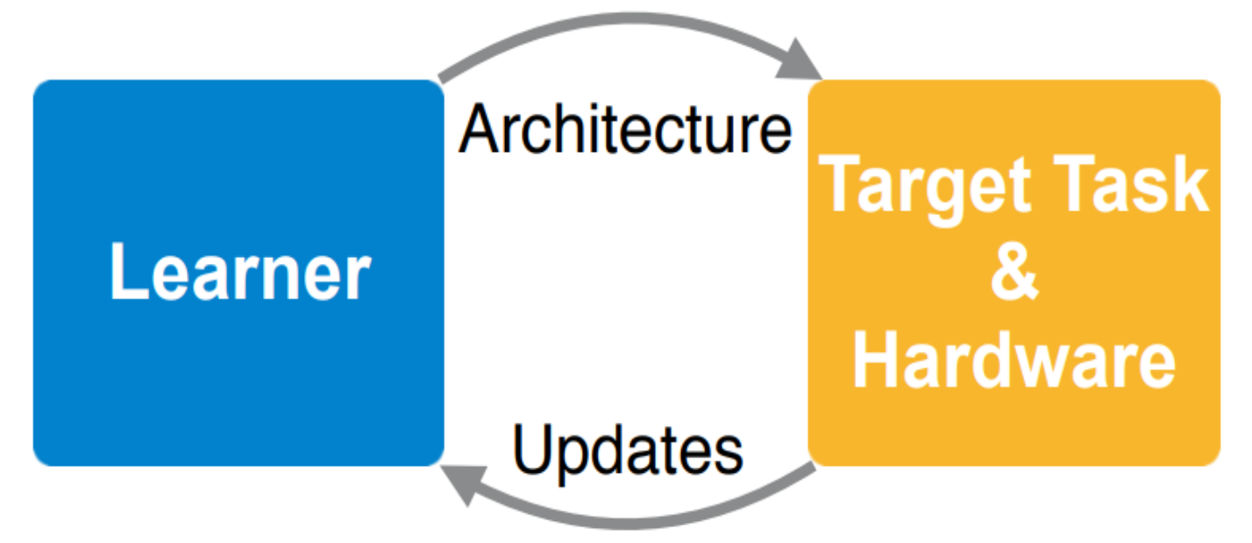
Different platform has different properties, e.g., degree of parallelism, cache size, memory bandwidth. We need to customize our models for each platform to achieve the best accuracy-efficiency trade-off.

Indirect Search to Direct Search

(1) Previous proxy-based approach



(2) Our proxy-less approach



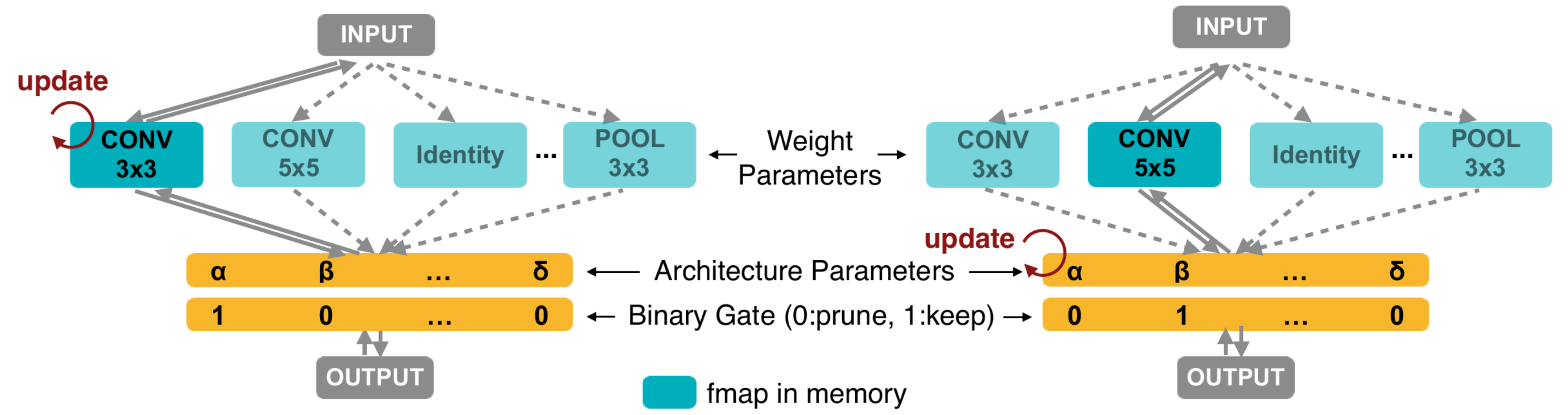
Conventional NAS is **VERY EXPENSIVE** (e.g., 48,000 GPU-hours) to run, thus relies on **proxy tasks** (e.g., CIFAR-10 -> ImageNet).

Goal: **Directly learn** neural network architectures on the large-scale target task and target hardware while allowing all blocks to have different structures.

Limitations of Proxy

- **Suboptimal for the target task**
- Blocks need to **share the same structure**
- Not optimize for the **target hardware**

Path-Level Pruning and Binarization



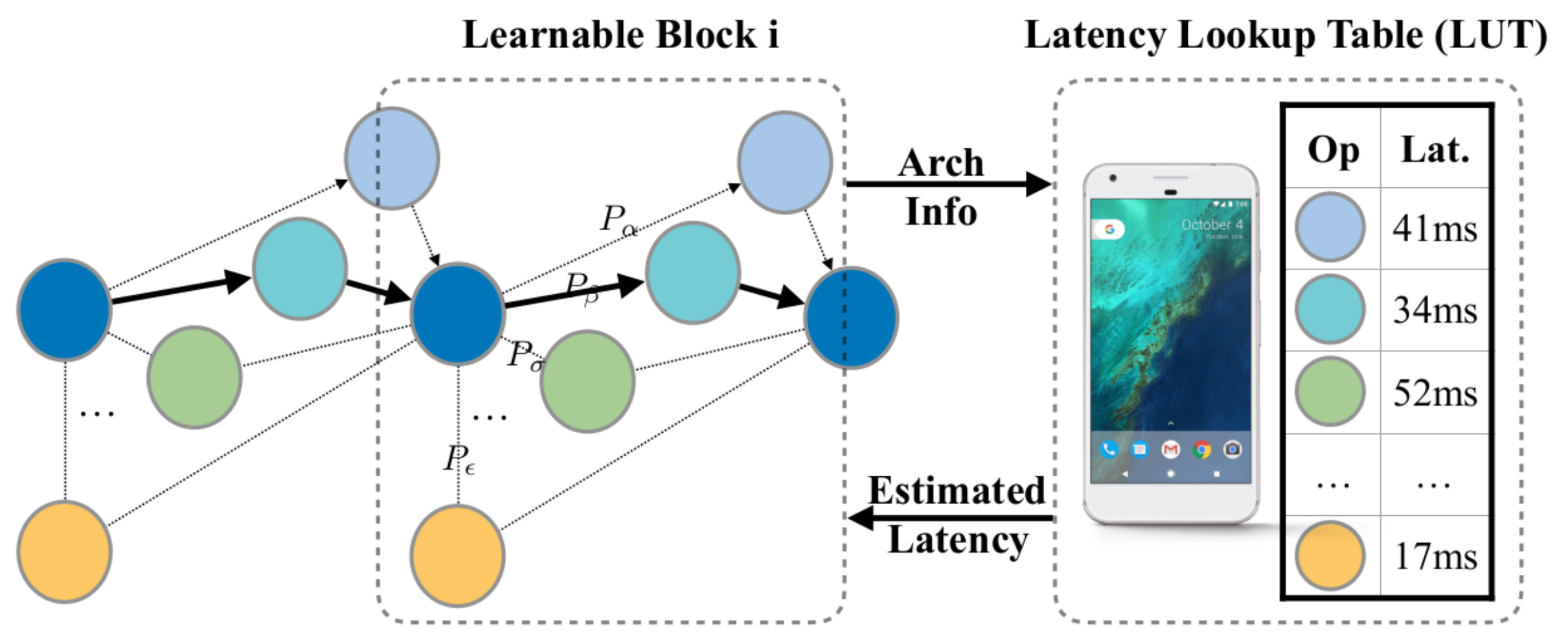
(1) Update weight parameters

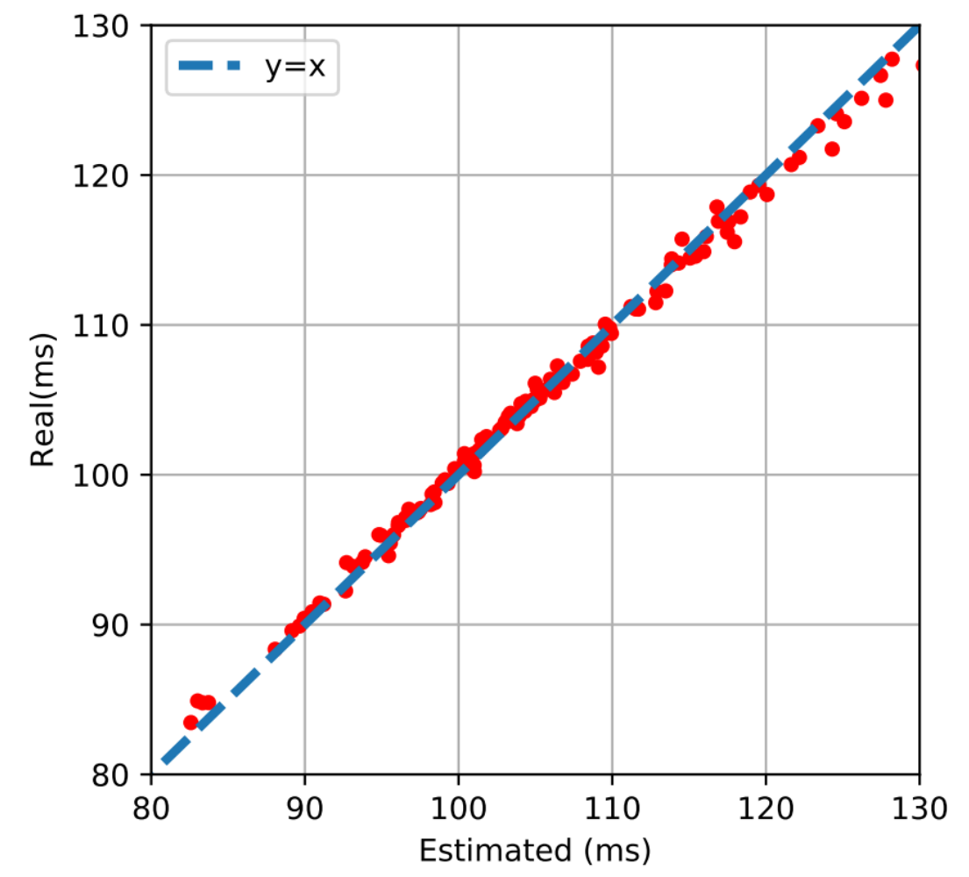
(2) Update architecture parameters

GPU hour-wise: Pruning redundant paths in a multi-path supernet.

GPU memory-wise: only one path of activation is active in memory at run-time.

Making Hardware Latency Differentiable





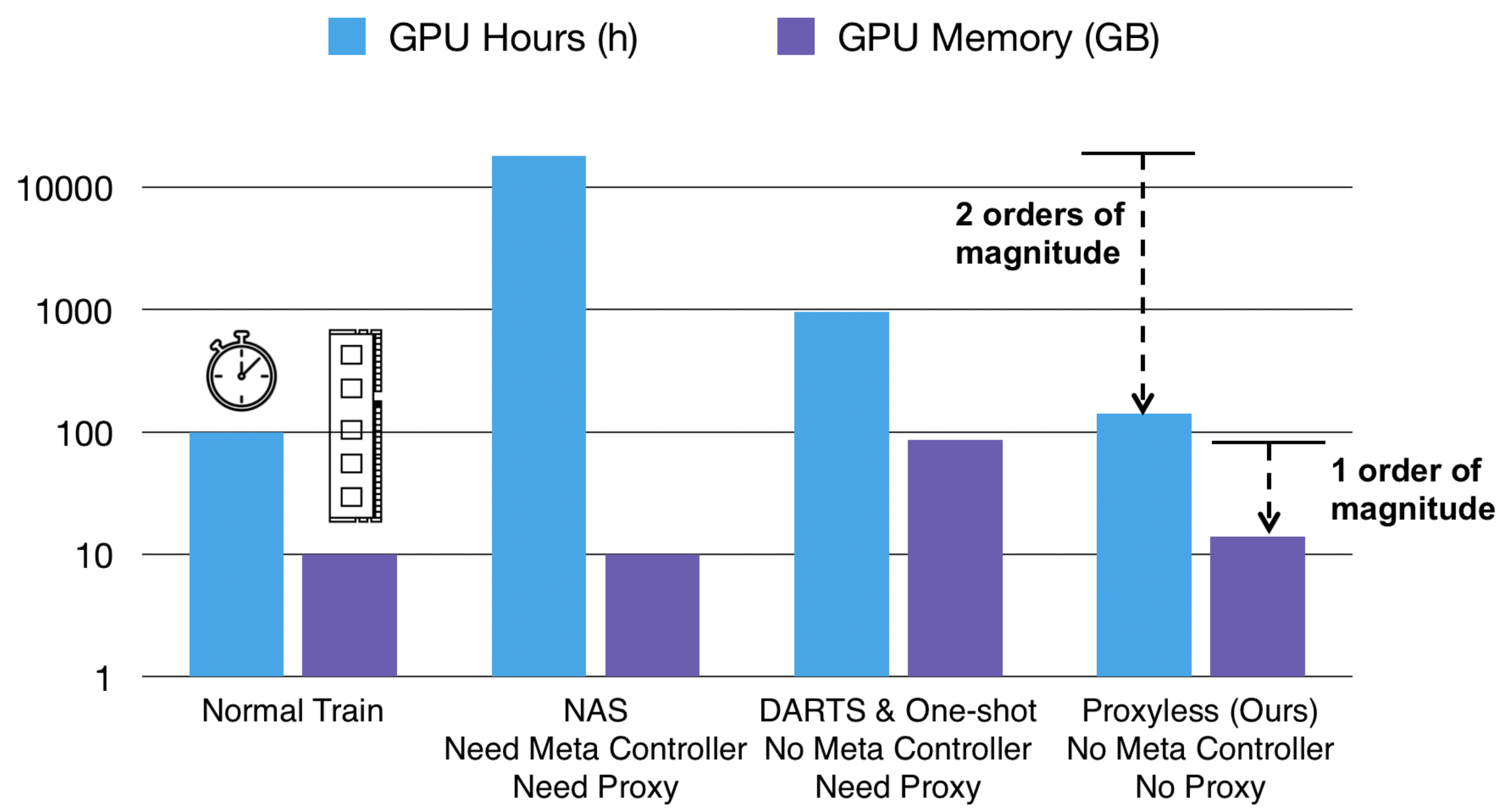
Expected latency is a continuous function of architecture parameters. We take the expected latency as a regularization term, thereby making latency differentiable.

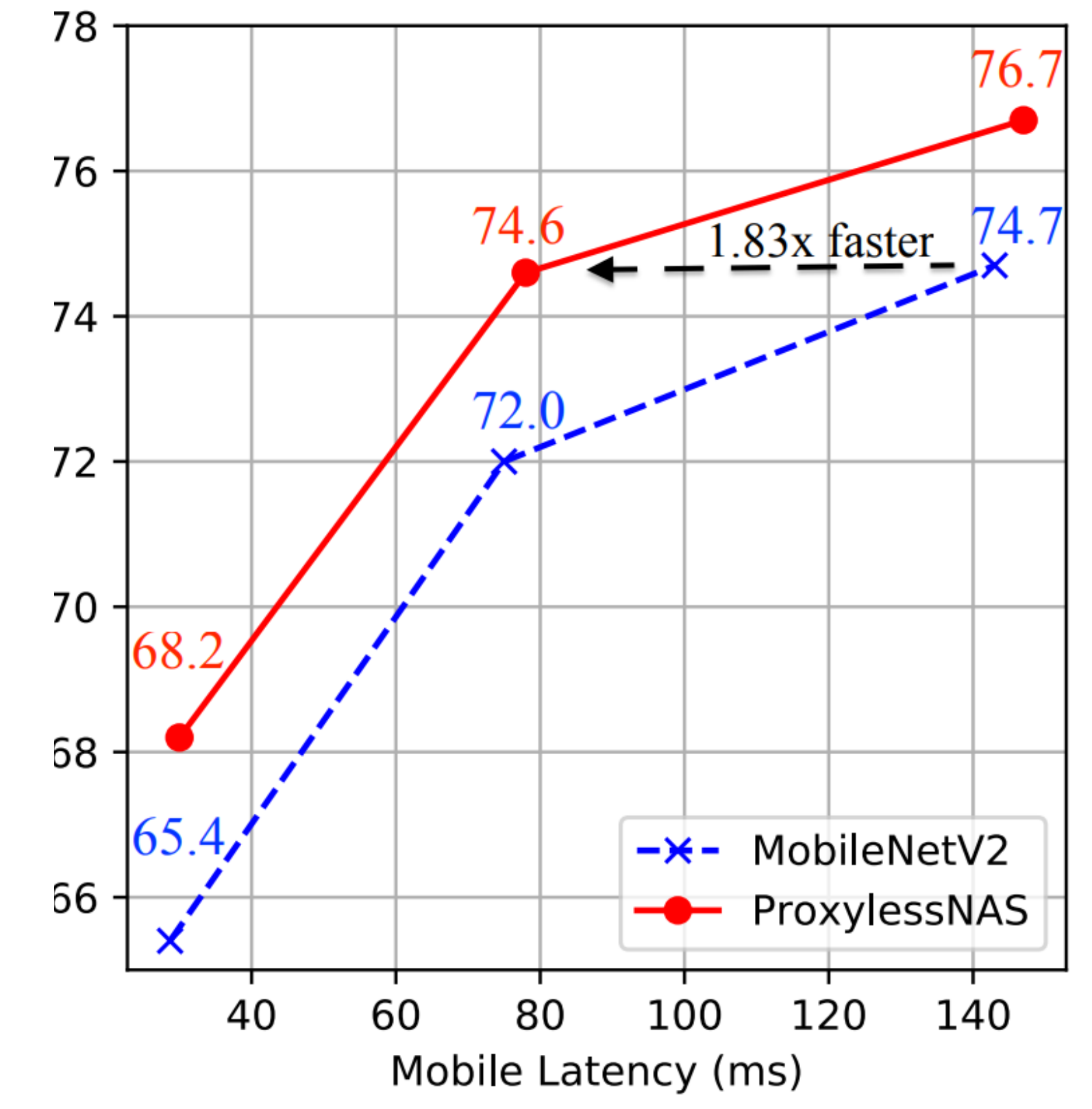
Results on ImageNet

Model	Top-1	Top-5	Mobile Latency	Hardware -aware	No Proxy	No Repeat	Search cost (GPU hours)
MobileNetV1 [16]	70.6	89.5	113ms	-	-	✗	Manual
MobileNetV2 [30]	72.0	91.0	75ms	-	-	✗	Manual
NASNet-A [38]	74.0	91.3	183ms	✗	✗	✗	48, 000
AmoebaNet-A [29]	74.5	92.0	190ms	✗	✗	✗	75, 600
MnasNet [31]	74.0	91.8	76ms	✓	✗	✗	40, 000
MnasNet (our impl.)	74.0	91.8	79ms	✓	✗	✗	40, 000
Proxyless-G (mobile)	71.8	90.3	83ms	✗	✓	✓	200
Proxyless-G + LL	74.2	91.7	79ms	✓	✓	✓	200
Proxyless-R (mobile)	**74.6**	**92.2**	78ms	✓	✓	✓	200

200x fewer

ProxylessNAS achieves state-of-the art accuracy (%) on ImageNet (under mobile latency constraint ≤ 80ms) with 200× less search cost in GPU hours.





The cost of ProxylessNAS is at the same level as regular training.

ProxylessNAS consistently outperforms MobileNetV2 under various latency settings. With the same level of top-1 accuracy as MobileNetV2 1.4, it runs 1.8× faster.

Model	Top-1	Top-5	GPU latency
MobileNetV2 (Sandler et al., 2018)	72.0	91.0	6.1ms
ShuffleNetV2 (1.5) (Ma et al., 2018)	72.6	-	7.3ms
ResNet-34 (He et al., 2016)	73.3	91.4	8.0ms
NASNet-A (Zoph et al., 2018)	74.0	91.3	38.3ms
DARTS (Liu et al., 2018c)	73.1	91.0	-
MnasNet (Tan et al., 2018)	74.0	91.8	6.1ms
Proxyless (GPU)	**75.1**	**92.5**	**5.1ms**

Our specialized model on GPU achieves 1.1% - 3.1% higher top-1 accuracy while being 1.2× faster, compared to MobileNetV2 and MnasNet.

Model	Top-1	GPU	CPU	Mobile
Specialized for GPU	75.1	**5.1ms**	204.9ms	124ms
Specialized for CPU	75.3	7.4ms	**138.7ms**	116ms
Specialized for Mobile	74.6	7.2ms	164.1ms	**78ms**

Hardware prefers specialized models.