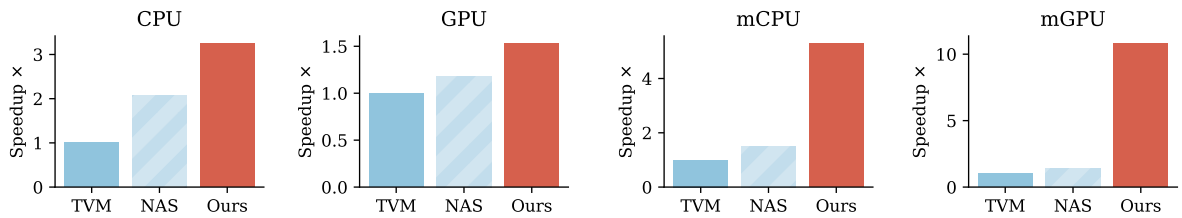
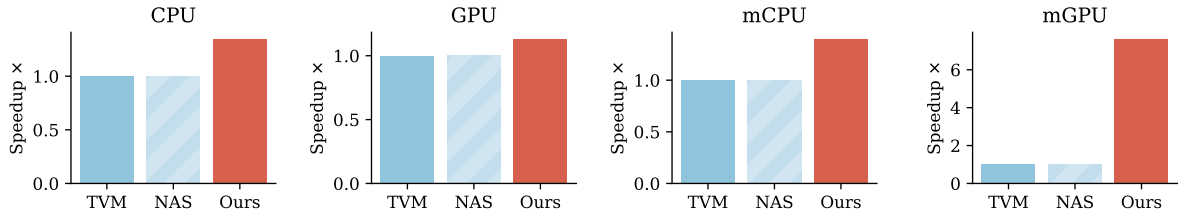
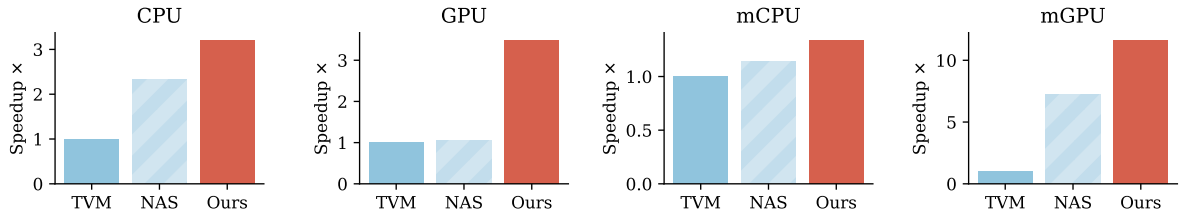


(a) ResNet-34



(b) ResNext-29-2x64d



(c) DenseNet-161

Figure 1: End-to-end performance for several networks on different hardware devices

architectures, from standard 3×3 convolutions in ResNet-34 to grouped convolutions in ResNeXt and a heavy reliance on 1×1 convolutions in DenseNet. We verify the safety of our compiler by training the output networks on two well known datasets: CIFAR-10 and ImageNet.

The networks are implemented with each operation written as a TVM Tensor Expression [2], which is an einsum-style syntax for expressing tensor computations. This is lowered to TVM IR, where our transformations can be employed. This allows for a fair comparison of our approach. Trained models and code are publicly available.

5. Key Results and Contributions

Our contributions are as follows:

1. We reformulate popular Neural Architecture Search techniques as *program transformations*. The NAS community describes different convolution types in ad hoc manner, ignoring structural equivalence while program transformation systems cannot reason about them.
2. We use Fisher Potential to provide transformation safety without the need to train. This means we can consider NAS search as program transformation exploration with standard compiler legality checks; *without any training in the search loop.*
3. We unify the transformation and architecture search spaces, *discovering new types of convolution*. We provide thorough analysis of three examples of new types of convolutional operators available in our framework that give significant performance improvement across networks and hardware.
4. We evaluate 3 networks using these operations, ResNet, ResNext and DenseNet, on 4 platforms and demonstrate, in most cases, more than $3\times$ inference speedup over a TVM baseline. In fact, in certain cases we achieve a $10\times$ improvement (see Figure 1). This is achieved without the extremely long search times of conventional NAS.
5. We are able to explore the accuracy/space co-design space of networks and find new designs including *a new Pareto optimal network*.

6. Why ASPLOS

Neural network efficiency is of great interest to the ASPLOS community. This paper directly connects programming languages and architecture by automatically determining how to transform a program to improve performance on a range of hardware. It also brings together two distinct communities — neural architecture search and compiler optimization — around the common goal of improved hardware performance.

References

[1] Paul Barham and Michael Isard. Machine learning systems are stuck in a rut. In *Proceedings of the Workshop on Hot Topics in Operating Systems*, pages 177–183. ACM, 2019.

[2] Tianqi Chen, Thierry Moreau, Ziheng Jiang, Lianmin Zheng, Eddie Yan, Haichen Shen, Meghan Cowan, Leyuan Wang, Yuwei Hu, Luis Ceze, et al. {TVM}: An automated end-to-end optimizing compiler for deep learning. In *USENIX Symposium on Operating Systems Design and Implementation*, 2018.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[4] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[5] Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. FBNet: Hardware-aware efficient convnet design via differentiable neural architecture search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[6] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[7] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.