Deterministic Data Structures for Efficient Embedded Deep Learning Architectures

Javier Fernandez-Marques[†], Sourav Bhattacharya[§], and Nicholas D. Lane[†§]
[†]University of Oxford, [§]Nokia Bell Labs

Since the success of AlexNet [Krizhevsky et al., 2012], convolutional neural networks (CNN) have become the preferred option for computer vision and audio applications. Recently, several approaches have been proposed to reduce the complexity of such systems making them deployable in constrained platforms such as wearables and other embedded devices. The majority of the existing research, excluding accelerators, can be subdivided into two categories:

- Layer Architectures: such as *bottleneck* [He et al., 2015] layers, that reduce the number of channels of the input tensor by using 1×1 filters prior to convolving it with a spatially larger kernel with the aim of reducing the number or of OPs. Along the same lines, depth-wise [Howard et al., 2017] convolutional layers split the standard convolutional layer into two convolutional layers reducing the number of OPs while maintaining high accuracy levels.
- Compression Techniques: They fixate their efforts in reducing the number of weights and in exploiting sparsity. Existing techniques such as *DeepCompression* [Han et al., 2015] is a conglomerate of clustering, quantisation and word encoding techniques. A step further is taken by [Wang et al., 2017] in which, for a given convolutional layer, filters are restricted to produce feature maps with minimal redundancy.

Our work introduces a novel technique to make complex CNN architectures be deployable on very constrained devices. Instead of directly learn the convolutional filters, we learn how to combine a set of deterministically codes that can be generated on-the-fly without any parameter. This makes it possible to design and deploy models without having store the filters of each layer as part of the model. In our first iteration of this idea [Tseng et al., 2018], we made use of a set of orthogonal binary codes that can be recursively generated on-the-fly in order to reconstruct the convolutional filters in our models. These codes are a base of the HD space where the filters lie. In our work we verify their suitability for image classification applications.

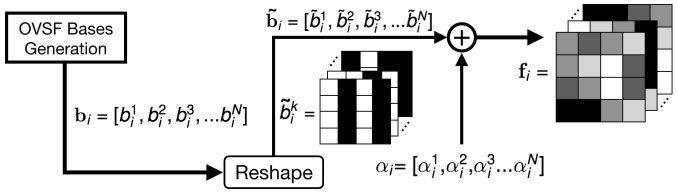

Figure 1: Filter generation by combining on-the-fly codes with a set of weights learnt.

For the next iteration of this idea, we are exploring new ways of introducing a higher ratio of deterministic or low parametrised elements into the network architecture aiming to ease the dependence on model parameters that need to be retrieved from disk/flash during inference, a serious bottleneck for applications that require low latency and minimal power consumption.

References

[Han et al., 2015] Han, S. et al. (2015). Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. *CoRR*, abs/1510.00149.

[He et al., 2015] He, K. et al. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

[Howard et al., 2017] Howard, A. G. et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861.

[Krizhevsky et al., 2012] Krizhevsky, A. et al. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

[Tseng et al., 2018] Tseng, V. W.-S., Bhattacharya, S., Marqués, J. F., Alizadeh, M., Tong, C., and Lane, N. D. (2018). Deterministic binary filters for convolutional neural networks. In *IJCAI*.

[Wang et al., 2017] Wang, Y. et al. (2017). Beyond filters: Compact feature map for portable deep model. In *ICML*.