# Few Shot Network Compression via Cross Distillation

Reading By, Harsh Srivastava
*B.Tech in Computer Science*
*IIT Patna*
Patna, India
harshsrivasta243@gmail.com

Guided By, Saurabh Sharma
*Ph.D, Computer Science*
*IIT Patna*
Patna, India

Supervised By, Dr. Joydeep Chandra
*Professor, Dept.of Computer Science*
*IIT Patna*
Patna, India

***Abstract*—This document is a (summarized) reading of the paper titled "Few Shot Network Compression via Cross Distillation" which is originally authored by Haoli Bai[1],Jiaxiang Wu[2], Irwin King[1] and Michael Lyu[1].**

**The Venue of Publication :**
**1. The Chinese University of Hong Kong**
**2. Tencent AI Lab [hlbai, king, lyu]@cse.cuhk.edu.hk, jonathanwu@tencent.com**
**The link to the original paper is here :**

**https://github.com/harshsrivasta243/1901CS24_CS299_2021/blob/Papers/Paper3.pdf**

## I. Motivations and Introduction

Model compression has been widely adopted to obtain lightweighted deep neural networks. Most prevalent methods, however, require fine-tuning with sufficient training data to ensure accuracy, which could be challenged by privacy and security issues. As a compromise between privacy and performance, in this reading we investigate few shot network compression.

Cross distillation is a novel layer-wise knowledge distillation approach. By interweaving hidden layers of teacher and student network, layer-wisely accumulated estimation errors can be effectively reduced. The proposed method offers a general framework compatible with prevalent network compression techniques such as pruning.

## II. Problem Addressed

This paper mainly aims at obtaining a compact student network from a highly parametrized teacher network.

## III. Baseline

To take care of security issues in network compression, some recent works motivate from knowledge distillation and propose data-free fine-tuning by constructing pseudo inputs from the pre-trained teacher network. However, these methods highly rely on the quality of the pseudo inputs and are therefore limited to small-scale problems.

Knowledge distillation is extended by minimizing layer-wise estimation errors (e.g., Euclidean distances) between the teacher and student network. The success of these approaches largely comes from the layer-wise supervision from the teacher network. Nevertheless, as there are few shot training samples available, the student network tend to over-fit on the training set and consequently suffer from high estimation errors from the teacher network during inference. Moreover, the estimation errors could propagate and accumulate layer-wisely and finally deteriorate the student network.

Cross distillation can effectively reduce the layer-wisely accumulated errors in the few shot setting, leading to a more powerful and generalizable student network. Specifically, to correct the errors accumulated in previous layers of the student network, we direct the teacher's hidden layers to the student network, which is called correction. Meanwhile, to make the teacher aware of the errors accumulated on the student network, we reverse the strategy by directing the student's hidden layers to the teacher network. With error-aware supervision from the teacher, the student can better mimic the teacher's behavior, which is called imitation. The correction and imitation compensate each other, and to find a proper trade-off, we propose to take convex combinations between either loss functions of the two procedures, or hidden layers of the two networks.

While most previous efforts on network compression rely on abundant training data for fine-tuning the compressed network, there is a recent trend on investigating security and privacy issues for network compression. These methods can be generally categorized into data-free methods and fewshot methods.

To perform data-free network compression, a simple way is to directly apply quantization or low-rank factorization on network parameters, which usually degrade the network significantly when the compression rate is high. Recent efforts motivate from knowledge distillation, which constructs pseudo inputs from the pre-trained teacher network based on its parameters, feature map statistics, or an independently trained

generative model to simulate the distribution of the original training set. However, the generation of high-quality pseudo inputs could be challenging and expensive, especially on large-scale problems.

Unlike data free compression techniques, few shot network compression can significantly improve the performance of the compressed network work with only limited training instances, which is potentially helpful for large-scale real-world problems. Our proposed cross distillation proceeds along the line of few shot network compression. As an extension of previous layer-wise regression methods, we pay extra attention to the reduction of estimation errors during inference, which are usually large as a result of over-fitting on few shot training instances. Similar ideas of cross connection between two networks are also previously explored in multitask learning to obtain mutual representations from different tasks.

## IV. The Novel technique of Cross Distillation

### A. Methods

Our goal is to obtain a compact student network $F^S$ from the over-parametrized teacher network $F^T$. Given few shot training instances $(x_n, y_n)^N{}_{n=1}$, we denote their corresponding l-th convolutional feature map of the teacher network $F^T$ as $h^T{}_l = \sigma(W^T{}_l * h^T{}_{l-1}) \in R^{(NXc_iXkXk)}$ where $\sigma(.)$ is the activation function, * is the convolutional operation, $W^T{}_l \in R^{(c_oXc_iXkXk)}$ is the 4-D convolutional kernel, and N, $c_o, c_i$,k are the number of training size, input channels, output channels and the kernel size respectively. Batch normalization layers are omitted as they can be readily fused into convolutional. Similar notations hold for $F^S{}_l$.

Unlike standard knowledge distillation approaches, here we adopt layer-wise knowledge distillation which can take layer-wise supervision from the teacher network. With previous layers being fixed, layerwise distillation aims to find the optimal $W^S{}_*$ that minimizes the Euclidean distance between $h^T$ and $h^S$, and depends on $L^T(W^S)$ which is called the estimation error and $R(W^S)$ which is some regularization tuned by $\lambda$.The student network $F^S$ tends to suffer from high estimation errors on the test set as a result of over-fitting. Moreover, the errors propagate and enlarge layer-wisely, and finally lead to a large performance drop on $F^S$.

### B. Overview

To address the above issue, we propose cross distillation, a novel layer-wise distillation method targeting at few shot network compression. Since the estimation errors are accumulated on the student network $F^S$ and $h^T$ are taken as the target during layer-wise distillation, we direct $h^T$ to $F^S$ in substitution of $h^S$ to reduce the historically accumulated errors.

However, directing $h^T$ to $F^S$ results in inconsistency because $F^S$ takes $h^T$ from $F^T$ in the training while it is expected to behave along during inference. In order to maintain the consistency and simultaneously make the teacher aware of the accumulated errors on the student net, we can inverse the strategy by guiding $h^S$ to $F^T$, which is also called imitation.

Despite the teacher network now can provide error-aware supervised signal, such connection brings inconsistency on the teacher network, which could be resolved by correcting the deviation due to layer-wise error propagation. Consequently, correction and imitation compensate each other and thus we find a proper balance between them by a convex combination of them, tuned by $\mu$, where $\mu \in [0,1]$

$$L = \mu * L^c + (1 - \mu) * L^i$$

where $L^c$ is the correction loss, $L^i$ is the imitation loss and L is their convex combination tuned by $\mu$ which is the net loss.

### C. Soft Cross Distillation

Although minimizing L is theoretically supported, the computation of ˜ L involves two loss terms with four convolutions to compute per batch of data, which doubles the training time. Hence we propose another variant to balance $L^c$ and $L^i$ by empirically soften the hard connection of $h^S$ and $h^T$. We define feature maps $\hat{}h^T$ and $\hat{}h^S$ after cross connection as the convex combination of $h^T$ and $h^S$, i.e.,

$$\begin{bmatrix} hT \\ hS \end{bmatrix} = \begin{bmatrix} \alpha & 1-\alpha \\ \beta & 1-\beta \end{bmatrix} \begin{bmatrix} h^T \\ h^S \end{bmatrix}$$

where $\alpha$, $\beta \in [0, 1]$ are the hyper-parameters that adjust how many percentages are used for cross connection. The convex combination ensures the norm of input to be nearly identical after cross connection and therefore parameter magnitude stays unchanged.

### D. Combined with Network Pruning

Cross distillation can be readily combined with a set of popular network compression techniques such as pruning or quantization, by taking different regularization $R(W^S)$. We linearly increase $\lambda$ to smoothly prune the student network, which empirically gives better results. This method works for network quantization as well. By taking $R(W^S)$ as the penalty to quantization points, our method can be combined with Straight Through Estimator (STE) or ProxQuant.

## V. Experiments on the Technique

### A. Methods Experimented

The main underlying, experimented methods that are put to test are structured pruning and unstructured pruning, both of which aim to reduce computational FLOPs and sizes of neural networks. Further we also analyse how cross distillation helps reduce the estimation error against varying size of the training set.

### B. Setup

Throughout the experiment, we use VGG and ResNet as base networks, and evaluations are performed on CIFAR-10 and ImageNet-ILSVRC12. All experiments are averaged over five runs with different random seeds, and results of means and standard deviations are reported.

### C. Baselines

For structured pruning, we compare our proposed methods against a number of baselines: 1) L1-norm pruning, a data-free approach; 2) Backpropagation (BP) based fine-tuning on L1-norm pruned models; 3) FitNet and 4) FSKD, both of which are knowledge distillation methods; 5) ThiNet and 6) Channel Pruning (CP), both of which are layer-wise regression based channel pruning methods. For unstructured pruning, we modify 1) to element-wise L1-norm based pruning. Besides, 4) FSKD, 5) ThiNet and 6) CP are removed since they are only applicable in channel pruning.

### D. Pruning Schemes

For the VGG- 16 network, we denote the three pruning schemes in the ascending order of sparsity as VGG-A, VGG-B and VGG-C respectively. We further prune half of the channels layer-wisely and denote the resulting scheme as VGG-50. For ResNet-34, we remove r per cent channels in the middle layer of the first three residual blocks with some sensitive layers skipped (e.g., layer 2, 8, 14, 16). The last residual block is kept untouched. The resulting structured pruning schemes are denoted as Res-r. Besides, we further remove 50 per cent channels for the last block to reduce more FLOPs when r = 70, denoted as Res-70.
In terms of unstructured pruning, we follow a similar pattern by removing r =50, 70, 90, 95 per cent parameters for both the VGG network and ResNet, and each layer is treated equally.

## VI. Results

### A. Structured Pruning

We evaluate structured pruning with VGG-16 on CIFAR-10 and ResNet-34 on ILSVRC-12. As the training size decreases, cross distillation brings more advantages comparing to the rest baselines, indicating that the layer-wise regression can benefit more from cross distillation when the student network over-fits more seriously on fewer training samples.
Next, we fix the training size and change the pruning schemes. We keep K = 5 on CIFAR-10 and K = 1 on ILSVRC-12.Again on both datasets our proposed cross distillation performs consistently better comparing to the rest approaches. Besides, the gain from cross distillation becomes larger as the sparsity of the student network increases (e.g., VGG-C and Res-70).We suspect that networks with sparser structures tend to suffer more from higher estimation errors, which poses more necessity for cross distillation to reduce the errors.

### B. Unstructured Pruning

Similar to structured pruning, we first fix the pruning scheme and vary the training size. The improvement is even larger comparing to structured pruning. One reason could be the irregular sparsity of network parameters cab better compensate the layer-wisely accumulated errors on $F^S$. We test our methods with different sparsities and hold the training size fixed as K = 1. As the sparsity r increases, cross distillation brings more improvement.

## VII. Further Analysis

Cross distillation brings the inconsistencies that could affect the reduction of estimation errors $L^r$. It can be observed that the student net trained by $L^c$ has large inconsistency, and the same for that trained by $L^i$. On the contrary, the student net trained by L shows lower inconsistency, and the estimation error $L^r$ is properly reduced as well. The results indicate that by properly controlling the magnitude of inconsistencies with soft connection, cross distillation can indeed reduce estimation errors $L^r$ and improve the student network.