Model Compression Using Optimal Transport

Suhas Lohit Michael Jones
Mitsubishi Electric Research Laboratories, Cambridge, MA, USA
slohit@merl.com mjones@merl.com

Abstract

Model compression methods are important to allow for easier deployment of deep learning models in compute, memory and energy-constrained environments such as mobile phones. Knowledge distillation is a class of model compression algorithms where knowledge from a large teacher network is transferred to a smaller student network thereby improving the student's performance. In this paper, we show how optimal transport-based loss functions can be used for training a student network which encourages learning student network parameters that help bring the distribution of student features closer to that of the teacher features. We present image classification results on CIFAR-100, SVHN and ImageNet and show that the proposed optimal transport loss functions perform comparably to or better than other loss functions.

1. Introduction

Deep convolutional neural networks are currently the most effective methods for many applications in computer vision including image classification and object detection [22]. However, network architectures yielding state-of-the-art network accuracies are memory and compute heavy [6]. Furthermore, recent evidence shows that it is necessary for neural networks to be overparameterized for gradient-based training procedures to be maximally effective [34, 3, 49]. This makes it difficult to employ them in resource-constrained environments such as mobile phones, drones etc. In order to overcome this disadvantage and reduce computational and memory requirements for inference, several model compression techniques have been shown to be of value. In particular, many knowledge distillation methods have been devised in order to improve the accuracy of smaller networks – student networks – by transferring the knowledge from more accurate larger neural networks – teacher networks [8]. Such approaches can reduce the network size by about $5\times$ with only a small drop in accuracy. The focus of this paper is however to develop a novel knowledge distillation approach by devising loss functions based on the optimal transport cost between teacher and student feature distributions.

As mentioned above, different classes of methods have been developed for model compression such as (a) pruning, where weights which do not affect the performance are removed [43, 38, 17], (b) training compact CNNs with sparsity constraints [27, 50, 44], (c) low-rank approximations of convolutional filters and weights [16, 45, 35, 20] and (d) knowledge distillation where the performance of a smaller student network is improved by transferring the knowledge from a larger more accurate teacher network [5, 19]. These methods can also be combined with each other for improved performance. Please read this excellent survey by Cheng et al. [8] for a more detailed treatment of these ideas.

In this paper, we consider an important class of knowledge distillation methods where a new loss function is used to train the student network in addition to the cross-entropy loss so as to encourage the student features to more closely match the teacher features. Optimal transport is a principled method of comparing two distributions which may not share support [42, 33]. We show that it is naturally applicable to the task of knowledge distillation and we design novel loss functions based on optimal transport tailored for the task of knowledge distillation. In our case, the two distributions are given by discrete measures of the student and teacher features, and the loss function encourages learning a student network that reduces the optimal transport cost between the two sets of features. We employ two different computationally efficient algorithms for computing the optimal transport cost on-the-fly while optimizing for the student network parameters. We conduct experiments on image classification on a variety of teacher-student network pairs using standard datasets and show that optimal transport-based losses help improve student network accuracies better than other comparable losses which match teacher and student features. We now review in more detail some closely related works in the literature.

2. Background and related work

2.1. Knowledge distillation methods

The earliest methods of knowledge distillation (KD) were developed by Buciluă et al. [5] and Hinton et al. [19] and have proven to be an effective method for transferring the knowledge learned by a complex teacher neural network into a simpler student network. Here, the basic idea is to use the softmax outputs of the last layer of the teacher network (i.e., soft labels) in the loss to train the student network, along with the usual cross-entropy loss to map to the one-hot vector given by the ground-truth label. Kullback-Leibler (KL) divergence, or a closely related variant, is used as the additional loss function between student outputs and the teacher's outputs, as both are probability distributions over the class labels and share support. In this paper, following recent works, we will refer to this loss as the KD loss. However, KD loss takes into account only the final outputs. Subsequent papers have looked at ways of exploiting intermediate layer outputs of the teacher network (i.e. from internal network layers) to further guide the training of the student network as in FitNets [36], attention transfer [48], relational knowledge distillation (RKD) [30], probabilistic knowledge transfer [31] etc. When combined with KD, all these methods result in significant improvements over KD alone. FitNets tries to match intermediate feature maps of teachers and students exactly, which may be too rigid of a constraint. On the other hand, RKD first computes pairwise relationships among teacher and student features in a batch and matches these geometric relationships between intermediate feature maps, rather than the feature maps themselves, which may be too loose of a constraint. Attention transfer (AT) [48] matches intermediate attention maps which are derived from intermediate feature maps.

In contrast to these methods, we propose a new loss for knowledge distillation based on optimal transport, to complement the standard KD loss. The central idea is to use optimal transport (OT) distance to measure the distance between the distributions of teacher and student feature maps in intermediate layers of the network. Another closely related work is Neuron Selectivity Transfer (NST) [21] which employs Maximum Mean Discrepancy (MMD) to measure the distance between the teacher and student feature sets. MMD is an alternative class of geometry-aware distances between distributions and is computed using the distance between kernelized versions of the teacher and student distributions. However, MMD suffers both from theoretical and practical drawbacks [13]. Theoretically, MMD does not faithfully capture the notion of distance in the original space. And practically, its gradient vanishes at the extreme points of the feature space. However, MMD is more easily scalable to large batches, compared to OT. We also note that recent methods also allow for interpolating between MMD and OT [13], however we do not investigate this direction in this paper. We also point readers to recent works by Menon et al. [28] and Zhou et al. [51] for a more theoretical treatment of knowledge distillation.

A recent paper based on the idea of using contrastive objectives [39] called Contrastive Representation Distillation [40] shows improvements over standard KD. The basic idea is to use a contrastive loss [39] to compare features from internal layers between a teacher and a student network. For every pair of teacher and student features corresponding to the same input image, referred to as a "positive" pair, a large set of "negative" teacher-student pairs is stored in memory. The authors then devise an algorithm to learn a student network that brings feature vectors in the positive pair closer to each other and drives feature vectors in negative student-teacher pairs away from each other. It achieves state-of-the-art accuracy on CIFAR-100. However, one of the drawbacks is that the method need access to large amounts of memory to store the large number of negative examples. We will refer to this loss as CRD. We consider this idea to be complementary to both KD, as well as various other matching losses like FitNets, NST and OT. That is, in principle, contrastive versions of FitNets and OT can also be developed and we consider this to be part of our future work. In our experiments, we show that combining OT + KD + CRD improves the state of the art for knowledge distillation for CIFAR-100. We also show results for optimal transport knowledge distillation using ImageNet and SVHN where we achieve results competitive with or better than the current state of the art.

2.2. Optimal transport for other applications in computer vision

Optimal transport is a principled, theoretically well-studied problem and is applicable to tasks where the notion of distance between two distributions is important [42]. The optimal transport problem was first studied in the late 18th century as a formalization of the problem of minimizing the cost of transporting resources to different locations. Around 1970, the concept of the Wasserstein distance (also known as the Earth Movers Distance) was proposed for a version of the optimal transport problem as a way of measuring the cost of transforming one probability distribution into another [12]. As a way of comparing probability distributions, the Wasserstein distance does not require the two distributions to have overlapping support unlike KL-divergence, for example. So in the case of comparing two distributions of feature vectors, if the distributions do not overlap, the KL-divergence would be infinity, while the Wasserstein distance would be a positive real number measuring the distance feature vectors in one distribution need to be moved to cover the feature vectors in the other.

In terms of applications to computer vision, optimal transport has previously been employed for domain adaptation, where the features of an unseen test domain are transformed into the training domain so as to achieve invariance to domain transformation [9]. In a similar vein, it has been applied to the problem of style transfer where the low-level features of a content image are transformed in such a way that they more closely match those of a style image [24]. Optimal transport loss functions have also been used to improve generative models as in the papers by Salimans et al. [37] and Genevay et al. [15]. We are the first to propose using optimal transport as a cost function for improved model compression.

3. Optimal transport for knowledge distillation

Given a trained teacher network and a student network which is to be trained using a gradient-based method, we pass a training set with b images through both the networks. Let the teacher features be $X = \{\mathbf{x}_1^{(l)}, \mathbf{x}_2^{(l)}, \ldots \mathbf{x}_b^{(l)}\}$ and the corresponding student features be $Y = \{\mathbf{y}_1^{(l)}, \mathbf{y}_2^{(l)}, \ldots \mathbf{y}_b^{(l)}\}$ for some intermediate layer l. Let $\mathbf{c}$ and $\hat{\mathbf{c}}$ be the ground-truth class-conditional distributions (one-hot encodings) and the corresponding predictions by the student network. The student network is trained by minimizing a combined loss function given by

$$L = L_{CE}(\mathbf{c}, \hat{\mathbf{c}}) + \alpha \sum_{l=1}^{l_{max}} L_{OT}(X^{(l)}, Y^{(l)}), \quad (1)$$

where $L_{CE}(\cdot,\cdot)$ is the usual cross-entropy loss used for classifcation and $L_{OT}(\cdot,\cdot)$ is the proposed optimal transport loss. l_{max} is the total number of layers at the which the sets of features are compared. In particular, we use the Wasserstein-1 metric, also called the Earth Movers Distance. $\alpha \in \mathbb{R}$ is used to balance the two losses. As we will discuss later, additional loss terms can be added to the above such as KD loss (L_{KD}) [19] and CRD loss [40] with appropriate weights. For example, we can combine L_{OT} and L_{KD} as

$$L = L_{CE}(\mathbf{c}, \hat{\mathbf{c}}_S) + \alpha \sum_{l=1}^{l_{max}} L_{OT}(X^{(l)}, Y^{(l)}) + \gamma L_{KD}(\hat{\mathbf{c}}_S, \hat{\mathbf{c}}_T). \quad (2)$$

Note that, in order to convert the given discrete feature sets into distributions, we use a uniformly scaled Dirac (unit mass) measure at each point. The optimal transport cost is given by

$$L_{OT}(X^{(l)}, Y^{(l)}) = \min_{T \geq 0} \sum_{i,j} T_{i,j}^{(l)} C_{i,j}^{(l)} \quad \text{s.t.} \sum_i T_{i,j}^{(l)} = \sum_j T_{i,j}^{(l)} = \frac{1}{b}, \quad (3)$$

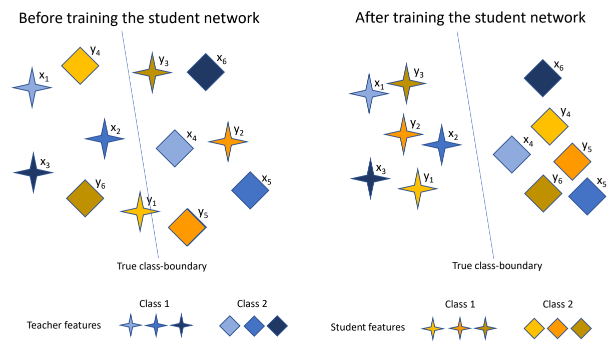

Figure 1. The figure illustrates the motivation for the proposed combination of cross-entropy loss with optimal transport for distillation. The student features $\mathbf{y}_j$'s get closer to the teacher features $\mathbf{x}_i$'s of the right class. Otherwise, the cross-entropy loss becomes high. However, note that $\mathbf{y}_j$ can be closest to $\mathbf{x}_i$, where $i \neq j$. This is indeed a good solution for maximizing student accuracy and the one-to-one corresponding alignment of teacher and student features is not a requirement. Optimal transport allows this solution and potentially results in greater flexibility when learning student features.

where $T^{(l)} \in \mathbb{R}^{b \times b}$ is called the transport matrix which encodes the soft-alignment between the two sets of vectors and C is the cost matrix which contains the distance between all teacher-student feature pairs at layer l. By $T^{(l)} \geq 0$, we mean that $\forall i, j, T_{i,j}^{(l)} \geq 0$. In the case of model compression, we want to learn the student parameters such that the optimal transport cost is reduced. This means that the student features should be geometrically close to the teacher features. In this paper, we employ the cosine distance on the unit sphere as the cost function. That is,

$$C_{i,j}^{(l)} = 1 - \frac{\mathbf{x}_i^{(l)T} \mathbf{y}_j^{(l)}}{\left\|\mathbf{x}_i^{(l)}\right\| \left\|\mathbf{y}_j^{(l)}\right\|}, \quad (4)$$

$\|\cdot\|$ is the Euclidean norm. We assume here that the intermediate features of both the teacher and the student have the same number of dimensions for all $l \in \{1, \ldots, l_{max}\}$. We relax this assumption later in some of the experiments.

Motivation and key insight: The OT loss function can be seen as a compromise between FitNets and RKD. It can be observed that this loss is less restrictive than FitNets [36] as it allows different examples fom the teacher features to be aligned to the student features as well as soft alignment, while it is more strict than RKD [30] as it requires the features to come closer to each other as measured by the distance function. To elaborate, the OT loss function is computed between two sets of features and is minimized when the two sets are equal **even if only up to a permutation**. As we are matching distributions of feature maps rather than directly matching feature maps, this constraint is less stringent than that used by FitNets. However, the constraint imposed by optimal transport is more stringent than the constraint imposed by RKD of merely matching pairwise geometric relationships computed separately in the teacher and student feature spaces. Our conjecture is that a loss based on optimal transport provides a better balance as a way of encouraging student feature maps to be similar to teacher feature maps. Our experiments will show that OT loss combined with KD loss improves over KD alone and also does better than FitNets + KD and RKD + KD. Figure 1 provides a clear illustration of the potential effect of the loss function presented in Equation 1 on the student features.

The expression for L_{OT} is computationally prohibitive to compute and scales cubically with b. Instead, we turn to two more tractable versions of the same expression, as discussed below.

3.1. Inexact Proximal Optimal Transport (IPOT)

A popular way of reducing the computational complexity of the OT problem is by adding an entropic regularization term [10], which we call regularized OT (ROT):

$$\begin{aligned} L_{ROT}(X^{(l)}, Y^{(l)}) &= \min_{T \geq 0} \sum_{i,j} T_{i,j}^{(l)} C_{i,j}^{(l)} + \epsilon h(T) \\ \text{s.t.} \sum_i T_{i,j}^{(l)} &= \sum_j T_{i,j}^{(l)} = \frac{1}{b}, \end{aligned} \tag{5}$$

where $h(T) = \sum_{i,j} T_{i,j} \log(T_{i,j})$ measures the entropy of the transportation matrix. The problem can then be solved efficiently using Sinkhorn iterations [4] which has been shown to scale close to quadratically with the batch size b [2]. However, the efficiency and numerical stability are sensitive to the choice of ϵ and in practice it is difficult to tune. Having too small a value ϵ leads to significantly larger number of Sinkhorn iterations needed for convergence, while having a large value of ϵ leads to numerical instability.

In this paper, we employ an improved version called the Inexact Proximal point method for exact Optimal Transport (IPOT) by Xie et al. [46] in order to compute L_{OT} and henceforth refer to this method as IPOT shown in Algorithm 3.1. Here, inexact proximal point iterations based on Bregman divergence are used. There exist works which show fast linear convergence under certain conditions for such methods. Xie et al. [46] show both theoretically and experimentally that the algorithm overcomes the drawbacks of earlier methods and converges to the exact solution with the same computational complexity as the Sinkhorn iterations. We note that it is easy to integrate algorithms to compute optimal transport with deep learning toolboxes with automatic differentiation and many dedicated libraries have also been developed for this purpose [14, 7].

Algorithm 1 Algorithm to compute IPOT(μ, ν, C) [46]

Input: The scaled Dirac masses μ, ν at the feature points in X, Y and the cost matrix C computed using Eqn. (4)
Output: The optimal transport cost $L_{OT} = \sum_{i,j} T^*_{i,j} C_{i,j}$
Initialize: $\mathbf{v} \leftarrow \frac{1}{b}\mathbf{1}_b, G_{i,j} = e^{-\frac{C_{i,j}}{\beta}}, T^{(0)} \leftarrow \mathbf{1}\mathbf{1}^T$
while $t < N$ **do**
  $Q \leftarrow G \odot T^{(t)}$, where $\odot$ is the element-wise product
  $\mathbf{u} \leftarrow \frac{\mu}{Q\mathbf{v}}, \mathbf{v} \leftarrow \frac{\nu}{Q^T\mathbf{u}}$
  $T^{(t+1)} \leftarrow \text{diag}(\mathbf{u}) Q \text{diag}(\mathbf{v})$
end while

3.2. Relaxed Earth-Mover's Distance (REMD)

As we will see from the experiments, although IPOT performs well, the training time can be quite long, depending on the number of Sinkhorn iterations. Instead, we consider a computationally efficient relaxation of the optimal transport problem following the work of Kusner et al. [26]. Instead of solving Equation (3), two simpler problems are first generated by dropping one set of constraints. We then have

$$\begin{aligned} R_{OT}^{(1)}(X^{(l)}, Y^{(l)}) &= \min_{T \geq 0} \sum_{i,j} T_{i,j}^{(l)} C_{i,j}^{(l)} \quad \text{s.t.} \sum_i T_{i,j}^{(l)} = \frac{1}{b} \\ R_{OT}^{(2)}(X^{(l)}, Y^{(l)}) &= \min_{T \geq 0} \sum_{i,j} T_{i,j}^{(l)} C_{i,j}^{(l)} \quad \text{s.t.} \sum_j T_{i,j}^{(l)} = \frac{1}{b}. \end{aligned} \tag{6}$$

The final relaxed EMD (REMD) is computed using

$$\begin{aligned} & L_{REMD}(X^{(l)}, Y^{(l)}) \\ &= \max(R_{OT}^{(1)}(X^{(l)}, Y^{(l)}), R_{OT}^{(2)}(X^{(l)}, Y^{(l)})) \\ &= \frac{1}{b} \max\left(\sum_i \min_j C_{i,j}^{(l)}, \sum_j \min_i C_{i,j}^{(l)} \right) \end{aligned} \tag{7}$$

4. Experimental results

4.1. CIFAR-100 [25]

This dataset[1] consists of 50000 training images and 10000 test images. Each image needs to be classified into 1 of 100 fine-grained categories. Each class has about 600 images in the dataset. All the images are in RGB format and are of size 32×32.

We use the same teacher and student combinations as in the paper by Tian et al. [40] as they provide a common

[1] https://www.cs.toronto.edu/ kriz/cifar.html

Teacher Student	WRN-40-2 WRN-16-2	resnet56 resnet20	resnet110 resnet20	resnet110 resnet32	resnet32x4 resnet8x4	vgg13 vgg8	resnet32x4 ShuffleNetV2
Teacher	75.61	72.34	74.31	74.31	79.42	74.64	79.42
Student (no distillation)	73.26	69.06	69.06	71.14	72.50	70.36	71.82
KD [19]	74.92	70.66	70.67	73.08	73.33	72.98	74.45
CRD [40]	**75.48**	**71.16**	**71.46**	**73.48**	**75.51**	**73.94**	**75.65**
CRD+KD	75.64	**71.63**	**71.56**	**73.75**	75.46	74.29	76.05
FitNet [36]	73.58	69.21	68.99	71.06	73.50	71.02	73.54
AT [48]	74.08	70.55	70.22	72.31	73.44	71.43	72.73
SP [41]	73.83	69.67	70.04	72.69	72.94	72.68	74.56
CC [32]	73.56	69.63	69.48	71.48	72.97	70.71	71.29
VID [1]	74.11	70.38	70.16	72.61	73.09	71.23	73.40
RKD [30]	73.35	69.61	69.25	71.82	71.90	71.48	73.21
PKT [31]	74.54	70.34	70.25	72.61	73.64	72.88	74.69
AB [18]	72.50	69.47	69.53	70.98	73.17	70.94	74.31
FT [23]	73.25	69.84	70.22	72.37	72.86	70.58	72.50
FSP [47]	72.91	n/a	69.95	71.89	72.62	70.23	n/a
NST [21]	73.68	69.60	69.53	71.96	73.30	71.53	74.68
REMD	74.19	70.66	70.76	72.96	73.97	72.45	73.58
REMD + KD	**75.79**	71.59	70.98	73.66	**76.06**	**74.35**	76.66
IPOT	74.79	71.04	70.79	72.87	74.19	72.80	73.97
IPOT + KD	75.63	71.32	71.29	73.68	75.99	74.29	**76.78**

Table 1. Comparison of various knowledge distillation methods on the CIFAR-100 dataset [19]. We are reporting the Top-1 accuracy (%) on the test set. We observe that the OT loss functions proposed in this paper – REMD and IPOT – generally outperform or are very close to all comparable methods which also measure similarity between internal teacher and student features such as FitNets, RKD etc. We find that IPOT leads to slightly better performance compared to REMD. All the numbers except for the OT losses are retrieved from [40]. Although using a contrastive objective i.e., CRD yields better results than other loss functions, OT + KD leads to the best results in several cases. The best results without KD are shown in **bold** and the best results with KD are shown in **red**. In row 2, "Student" refers to the student networks trained without using a distillation loss. The numbers for the OT losses are averaged over 3 runs and the standard deviation are shown in Table 4.

framework for benchmarking various knowledge distillation methods. The teacher and student networks on which we conduct experiments, the number of weights and the corresponding compression ratios are given in the supplement.

Implementation details: All the hyperparameters are held constant for all the methods, We use a batch size of 64 and train for 240 epochs. An initial learning rate of 0.05 is used and reduced by multiplying by $\frac{1}{10}$ after $150, 180, 210$ epochs. We use Top-1 accuracy on the test set to compare the performance between various distillation methods. We compare our method with several other recent and popular knowledge distillation methods that also encourage similarity between teacher and student features at different layers including FitNets [36], RKD [30], AT [48], SP [41], CC [32], PKT [31], AB [18], FT [23], FSP [47] and NST [21], as well as KD [19] and CRD [40].

We report results on two methods of calculating the OT distance: REMD and IPOT and furthermore, the combination of the OT losses with KD and CRD. All the student networks are divided into 4 stages of approximately equal depth, independent of the architecture or depth of the overall network. The output of the fourth stage is the penultimate layer output, immediately before the softmax operation. During training of student networks, along with the softmax outputs required to calculate the cross-entropy loss, features at the outputs of these 4 intermediate stages for both student and teacher features are extracted. Then OT losses can be computed for one or more of four sets of features between the student and teacher which is also the procedure used for all the baseline algorithms. The number of stages used for loss computation is a hyperparameter. For all of our main experiments, we compute and add the OT loss between all four sets. Later, as an ablative study, we show that even computing the OT loss at a single layer can yield close to optimal performance.

Also note that when the teacher and student architectures are different as in the case of resnet32x4/ShuffleNetV2, we use additional embedding layers which map from the teacher space to the student space, following FitNets [36] and CRD [40]. In particular, if the features being compared are from the penultimate layer, we use separate linear layers to map both the teacher and student features to a common dimensional space. When the features being compared are derived from intermediate convolutional layers, we use additional convolutional layers to map the teacher features to the same dimensionality and shape as the student features. In our experiment with resnet32x4/ShuffleNetV2, the embedding dimensionality for the penultimate layer is set to 128.

When adding the OT loss to cross-entropy, we experimented with two weights for the OT loss (see Eq. (1)), $\alpha = 0.9, 1.0$ and we report the better of the two results in the main paper based on a validation set. Accuracies obtained for both values for IPOT are provided in Table 2. When using IPOT, we employ $\beta = 20$ following Xie et al. [46] and the number of proximal point iterations, $N = 50$, as it provides a good balance between training time and accuracy. In our experiments on an Nvidia GTX 1080, REMD and IPOT take about 2.5 hours and 10 hours, respectively, for training.

Features used for loss computation	IPOT	IPOT + KD
Output of stage 1	73.59	75.20
Output of stage 2	74.01	75.39
Output of stage 3	74.25	75.53
Output of stage 4	73.27	75.66
Using outputs of all 4 stages	74.19	75.99

Table 2. Contribution of IPOT loss to performance per layer used for loss computation. We report the Top-1 accuracy (%) on the CIFAR-100 test set using resnet32x4 teacher and resnet8x4 student. Note that resnet8x4 without distillation yields 72.50% accuracy.

Results: All the main results for CIFAR-100 are shown in Tables 1 and 3. The results for the baseline methods are taken from the paper by Tian et al. [40]. Table 1 contains the Top-1 accuracy on the CIFAR-100 test using various knowledge distillation methods for various pairs of teacher-student architecture pairs. We make the observation that both REMD and IPOT either outperform or yield similar performance to all comparable baselines with the exception of CRD. Note that CRD uses a contrastive objective, unlike all the other methods. More importantly, in Table 3, we show results when we combine various loss functions with the KD loss. Again, we see similar trends as before. However, the boosts in performance are more pronounced for both REMD+KD and IPOT+KD, which outperform all the baseline loss functions, including CRD+KD, for nearly all teacher-student pairs. We also show that we can obtain further improved results by combining IPOT+CRD+KD. For this setting, we use an equal weight of 1.0 for all the four loss terms. This is just an illustration that a combination of more loss terms can result in better performance. This may occur with other combinations of other loss functions. However, conducting such a large experiment to find the optimal combination is beyond the scope of this paper.

Ablation study: In the experiments above, we use the student and teacher features from all four layers to compute the OT loss. Here, we find out their individual contribution to the final performance when using IPOT, as shown in Table 2. Surprisingly, we find that even though using all layers yields better performance, even computing the OT loss on a single layer yields nearly as good performance for this dataset. It is possible that combining specific pairs or triplets of layers rather than using all layers, may lead to even better performance.

Effect of α and γ for IPOT-based loss functions: The loss function used to train the student networks is given in Equation (2). It shows that, in addition to the usual cross-entropy loss $L_{CE}(\cdot, \cdot)$ (between the ground-truth labels written as a one-hot encoding $\mathbf{c}$ and the output of the student network $\hat{\mathbf{c}}$), we add the optimal transport (OT) loss proposed in this paper, weighted by α, and the KD loss [19] weighted by γ. For the experiment on CIFAR-100, we investigate the effect of α and γ, and the results for different teacher-student pairs are shown in Table 4. All the numbers show the top-1 accuracy on the test set averaged over 3 runs. Additionally, the table also shows the standard deviation for the 3 runs.

Transportation matrix from using the optimal transport loss: We conduct the following analysis on the trained models on CIFAR-100 using the proposed IPOT loss function for training the student models. For each teacher-student pair, we compute the final layer features for all the images in the test set (10000 images) throughout the training process. We then compute the transportation matrix T using IPOT. T is of size 10000×10000, and thus, cannot be visualized easily. Instead, we compute the trace $tr(T) = \sum_i T_{i,i}$. Note that T is normalized appropriately such that, if each student feature is aligned exactly with the corresponding teacher feature, $tr(T) = 1.0$. Figure 2 shows the evolution of $tr(T)$ as the student network is trained. We see that, even when the training is finished, the student features are not exactly aligned with the corresponding teacher features. At the same time, the test set performance produced by this method is higher than most other KD methods. Particularly surprising are the results of resnet32x4/resnet8x4 and WRN-40-2/WRN-16-2 where the final $tr(T)$ is only about 0.88. This can be interpreted as, on average, 12% of the student features not being exactly aligned with the corresponding teacher features, but

Teacher Student	WRN-40-2 WRN-16-2	resnet110 resnet20	resnet32x4 resnet8x4	vgg13 vgg8	resnet32x4 ShuffleNetV2
Teacher	75.61	74.31	79.42	74.64	79.42
Student (no distillation)	73.26	69.06	72.50	70.36	71.82
KD [19]	74.92	70.67	73.33	72.98	74.45
CRD+KD [40]	75.64	**71.56**	75.46	74.29	76.05
FitNet+KD [36]	75.12	70.67	74.66	73.22	75.15
AT+KD [48]	75.32	70.97	74.53	73.48	75.39
SP+KD [41]	74.98	71.02	74.02	73.49	74.88
CC+KD [32]	75.09	70.88	74.21	73.04	74.71
VID+KD [1]	75.14	71.10	74.56	73.19	74.85
RKD+KD [30]	74.89	70.77	73.79	72.97	74.55
PKT+KD [31]	75.33	70.72	74.23	73.25	74.66
AB+KD [18]	70.27	70.97	74.40	73.35	74.99
FT+KD [23]	75.15	70.88	74.62	73.44	75.06
NST+KD [21]	74.67	71.01	74.28	73.33	75.24
REMD + KD	**75.79**	70.98	**76.06**	**74.35**	76.66
IPOT + KD	75.63	71.29	75.99	74.29	**76.78**
IPOT + CRD	75.57	71.47	76.06	74.30	76.81
IPOT + CRD + KD	**76.22**	**71.81**	**76.82**	**74.79**	**76.81**

Table 3. Comparison of various knowledge distillation methods on the CIFAR-100 dataset when combined with KD loss [19]. We are reporting the Top-1 accuracy (%) on the test set. We can easily observe that the OT loss functions proposed in this paper – REMD+KD and IPOT+KD – outperform all comparable methods which measure similarity between internal teacher and student features. All the numbers except for the OT losses are retrieved from [40]. We also see that CRD + KD, which proposes a contrastive objective produces similar performance as REMD+KD and IPOT+KD. The best results are shown in **red** and the second best results are shown in **bold**. In row 2, "Student" refers to the student networks trained with only the usual cross-entropy loss and without using a distillation loss. The numbers for the OT losses are averaged over 3 runs and the standard deviation are shown in Table 4.

Teacher / Student Weights	WRN-40-2 WRN-16-2	resnet56 resnet20	resnet110 resnet20	resnet110 resnet32	resnet32x4 resnet8x4	vgg13 vgg8	resnet32x4 ShuffleNetV2
$\alpha = 0.9, \gamma = 0.0$	74.17 ± 0.62	71.04 ± 0.32	70.79 ± 0.39	72.87 ± 0.06	74.19 ± 0.50	72.75 ± 0.15	73.06 ± 0.20
$\alpha = 1.0, \gamma = 0.0$	74.79 ± 0.09	70.64 ± 0.13	70.67 ± 0.05	72.66 ± 0.07	74.03 ± 0.18	72.80 ± 0.29	73.97 ± 0.62
$\alpha = 0.9, \gamma = 1.0$	75.63 ± 0.21	71.21 ± 0.28	71.29 ± 0.24	73.54 ± 0.25	75.99 ± 0.04	74.29 ± 0.31	76.78 ± 0.12
$\alpha = 1.0, \gamma = 1.0$	75.35 ± 0.20	71.32 ± 0.22	71.18 ± 0.54	73.68 ± 0.57	75.88 ± 0.41	74.21 ± 0.36	76.37 ± 0.51

Table 4. Effect of α and γ on the Top-1 accuracy for CIFAR-100 classification for IPOT and KD, and the corresponding standard deviation over 3 runs.

more with the features of other images, as shown in Figure 1 of main paper.

4.2. Imagenet [11]

The ILSVRC2012 dataset[2], also called the ImageNet dataset is a large database containing 1.2 million RGB images in the training set and 50000 RGB images in the validation set. The task is image classification and each image belongs to one of 1000 categories. As the test set is not available, the validation set is itself employed as the test set. All the images are resized to 224×224 for training. We con-

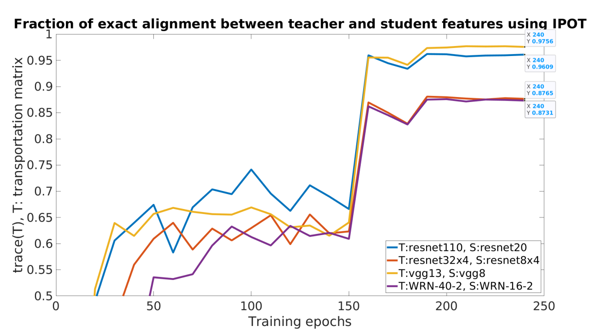

Figure 2. $tr(T)$ during training.

[2]http://www.image-net.org/download-images

	Teacher	Student	KD	Online KD *	CRD	CRD+KD	AT	SP	CC	IPOT	IPOT+KD
Top-1	26.69	30.25	29.34	29.45	28.83	28.62	29.30	29.38	30.04	29.54	28.88
Top-5	8.58	10.93	10.12	10.41	9.87	9.51	10.00	10.20	10.83	10.48	9.66

Table 5. Top-1 and Top-5 error rates (%) on ImageNet validation set for different knowledge distillation losses. The teacher network is ResNet-34 and the student network is ResNet-18. IPOT+KD performs close to the state-of-the art which is CRD+KD. "Student" refers to the student network trained with only the usual cross-entropy loss and without using a distillation loss.

T-S pair	Teacher	Student	KD	CRD	CRD+KD	FitNet	Fitnet+KD	RKD	RKD+KD	PKT	PKT+KD	REMD	REMD+KD	IPOT	IPOT+KD
resnet32x4 resnet8x4	94.36	90.39	94.49	**94.96**	**95.47**	91.32	94.48	93.30	94.58	90.77	94.38	89.66	94.49	91.63	94.73
WRN-40-2 WRN-16-2	94.52	93.45	95.22	94.74	95.25	93.93	95.27	95.23	**95.39**	93.68	95.15	93.15	94.94	94.28	**95.41**

Table 6. Accuracy (%) on the SVHN test set for different knowledge distillation losses for two teacher-student (T-S) pairs. IPOT+KD yields state-of-the-art performance for WRN-40-2/WRN-16-2 and the second-best results for resnet32x4/resnet8x4. The best results are shown in **red** and the second best results are shown in **bold**. "Student" refers to the student networks trained with only the usual cross-entropy loss and without using a distillation loss.

duct our experiment comparing the proposed optimal transport loss using IPOT with other loss functions which are applied at the intermediate layers including AT, SP and CC, as well as KD and CRD. A pretrained ResNet-34 released by PyTorch serves as the teacher network while a ResNet-18 is used as the student network. The results are shown in Table 5 where we report the Top-1 and Top-5 error percentages on the validation set for all the methods. We see that IPOT loss leads to comparable performance with other loss functions while CRD clearly performs the best. When IPOT is combined with KD, the top-1 error rate decreases to 28.88% which is comparable to the 28.62% top-1 error rate using CRD+KD. Note that the all the results except those involving IPOT loss function are reported as in the paper by Tian et al. [40].

4.3. SVHN dataset [29]

The Street View House Numbers (SVHN) [3] contains images belonging to 10 classes consisting of 10 digits 0-9 cropped from real world images of house numbers. The images are of size 32×32 and have RGB channels. The training set contains 73257 images and the test set contains 26032 images. We perform a similar set of experiments as in the case of CIFAR-100. We use two pairs of teacher-student networks and for training the student networks, we use a batch size of 200, an initial learning rate of 0.001 and 150 epochs for training the networks. The learning rate is reduced by multiplying by $\frac{1}{10}$ after $80, 110, 135$ epochs. The remaining hyperparameters are identical to those employed for the experiments on CIFAR-100. We compare IPOT and REMD with KD, FitNets, RKD, PKT and CRD for two sets of teacher-student pairs. The results thus obtained are shown in Table 6. We immediately see that IPOT outperforms REMD in all cases. Overall, IPOT yields better results than other loss functions except CRD and RKD for this dataset. However, when KD is added to all the loss functions, IPOT+KD yields state-of-the-art performance for WRN-40-2/WRN-16-2 and the second-best results for resnet32x4/resnet8x4.

[3]http://ufldl.stanford.edu/housenumbers/

5. Discussion

In this paper, we proposed loss functions for model compression based on optimal transport. The loss functions measure the distance between the distribution of features from a teacher network and the distribution of student features. Minimizing these losses while training student networks encourages them to learn features that belong to the same distribution as the teacher. We verified experimentally on CIFAR-100, ImageNet and SVHN that the proposed losses perform better or as well as various other loss functions proposed in the KD literature. This is true across many teacher-student architecture pairs as well. This work suggests a future direction in developing "contrastive" optimal transport loss functions for knowledge distillation, following the work of [39], is a promising direction. The loss functions can encourage bringing closer teacher and student distributions for positive pairs and push farther apart distributions for negative pairs. These losses could further take into account the semantic content in the training images while creating positive and negative pairs. (CRD only labels a teacher/student feature pair as positive if it comes from the same image, as opposed to coming from the same object class.) Another interesting avenue involves developing and employing faster methods for comparing distributions such as using interpolated distances between MMD and OT which combines advantages of both methods [13].

References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9163–9171, 2019.

[2] Jason Altschuler, Jonathan Niles-Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *Advances in Neural Information Processing Systems*, pages 1964–1974, 2017.

[3] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[4] Jean-David Benamou, Guillaume Carlier, Marco Cuturi, Luca Nenna, and Gabriel Peyré. Iterative bregman projections for regularized transportation problems. *SIAM Journal on Scientific Computing*, 37(2):A1111–A1138, 2015.

[5] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 535–541, 2006.

[6] Alfredo Canziani, Adam Paszke, and Eugenio Culurciello. An analysis of deep neural network models for practical applications. *arXiv preprint arXiv:1605.07678*, 2016.

[7] Benjamin Charlier, Jean Feydy, Joan Glaunès, François-David Collin, and Ghislain Durif. Kernel operations on gpu, without memory overflows kernel operations on the gpu, with autodiff, without memory overflows. *arXiv preprint arXiv:2004.11127*, 2020.

[8] Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *IEEE Signal Processing Magazine*, 2019.

[9] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

[10] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[12] Roland Dobrushin. Prescribing a system of random variables by conditional distributions. *Theory of Probability and its Applications*, 15(3), 1970.

[13] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. Interpolating between optimal transport and mmd using Sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690, 2019.

[14] R'emi Flamary and Nicolas Courty. Pot python optimal transport library, 2017.

[15] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with Sinkhorn divergences. *arXiv preprint arXiv:1706.00292*, 2017.

[16] Yunchao Gong, Liu Liu, Ming Yang, and Lubomir Bourdev. Compressing deep convolutional networks using vector quantization. In *arXiv:1412.6115*, 2014.

[17] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, 2015.

[18] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3779–3787, 2019.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[20] Lu Hou, Quanming Yao, and James T. Kwok. Loss-aware binarization of deep networks. In *International Conference on Learning Representations*, 2016.

[21] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017.

[22] Asifullah Khan, Anabia Sohail, Umme Zahoora, and Aqsa Saeed Qureshi. A survey of the recent architectures of deep convolutional neural networks. *arXiv preprint arXiv:1901.06032*, 2019.

[23] Jangho Kim, SeongUk Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. In *Advances in Neural Information Processing Systems*, pages 2760–2769, 2018.

[24] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10051–10060, 2019.

[25] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical Report TR-2009, University of Toronto, 2009.

[26] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966, 2015.

[27] Vadim Lebedev and Victor Lempitsky. Fast convnets using group-wise brain damage. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[28] Aditya K Menon, Ankit Singh Rawat, Sashank Reddi, Seungyeon Kim, and Sanjiv Kumar. A statistical perspective on distillation. In *International Conference on Machine Learning*, 2021.

[29] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NeurIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[30] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019.

[31] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 268–284, 2018.

[32] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5007–5016, 2019.

[33] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[34] Tomaso Poggio, Qianli Liao, Brando Miranda, Andrzej Banburski, Xavier Boix, and Jack Hidary. Theory iiib: Generalization in deep networks. Technical Report CBMM Memo No. 90, MIT Center for Brains, Minds and Machines, June 2018.

[35] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[36] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[37] Tim Salimans, Han Zhang, Alec Radford, and Dimitris Metaxas. Improving gans using optimal transport. *International Conference on Learning Representations*, 2018.

[38] Suraj Srinivas and Venkatesh Babu. Data-free parameter pruning for deep neural networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2015.

[39] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *International Conference on Machine Learning Workshop on Self-supervised Learning*, 2019.

[40] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *International Conference on Learning Representations*, 2020.

[41] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1365–1374, 2019.

[42] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[43] Huan Wang, Qiming Zhang, Yuehai Wang, and Haoji Hu. Structured probabilistic pruning for deep convolutional neural network acceleration. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[44] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, 2016.

[45] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qianghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[46] Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. A fast proximal point method for computing exact wasserstein distance. *arXiv preprint arXiv:1802.04307*, 2018.

[47] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4133–4141, 2017.

[48] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[49] Chiyuan Zhang, Samy Bengio, Benjamin Hardt, Moritz nd Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *Proceedings of the International Conference on Learning Representations*, 2017.

[50] Hao Zhou, Jose M. Alvarez, and Fatih Porikli. Less is more: Towards compact CNNs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[51] Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *International Conference on Learning Representatons*, 2021.