

# Estimating and Maximizing Mutual Information for Knowledge Distillation

Aman Shrivastava<sup>1</sup>   Yanzun Qi<sup>1</sup>   Vicente Ordonez<sup>2</sup>  
<sup>1</sup>University of Virginia, <sup>2</sup>Rice University  
 {as3ek, yq2h}@virginia.edu, vicenteor@rice.edu

## Abstract

In this work, we propose *Mutual Information Maximization Knowledge Distillation (MIMKD)*. Our method uses a contrastive objective to simultaneously estimate and maximize a lower bound on the mutual information of local and global feature representations between a teacher and a student network. We demonstrate through extensive experiments that this can be used to improve the performance of low capacity models by transferring knowledge from more performant but computationally expensive models. This can be used to produce better models that can be run on devices with low computational resources. Our method is flexible, we can distill knowledge from teachers with arbitrary network architectures to arbitrary student networks. Our empirical results show that MIMKD outperforms competing approaches across a wide range of student-teacher pairs with different capacities, with different architectures, and when student networks are with extremely low capacity. We are able to obtain 74.55% accuracy on CIFAR100 with a ShuffleNetV2 from a baseline accuracy of 69.8% by distilling knowledge from ResNet-50. On Imagenet we improve a ResNet-18 network from 68.88% to 70.32% accuracy (1.44%+) using a ResNet-34 teacher network.

## 1. Introduction

Recent machine learning literature has seen a lot of progress driven by deep neural networks. Many such models that achieve state-of-the-art performance on different benchmarks require large amounts of computation and memory capacities [14]. These requirements limit the wider adoption of these models in resource-limited scenarios. To this end, Knowledge Distillation (KD) has been used to transfer knowledge from a stronger teacher network to a smaller and less computationally expensive student network [4, 12]. This often allows student networks to outperform identical models trained without a teacher. However there is still much room for improvement in knowledge distillation so that students can extract as much knowledge as possible from the teacher network.

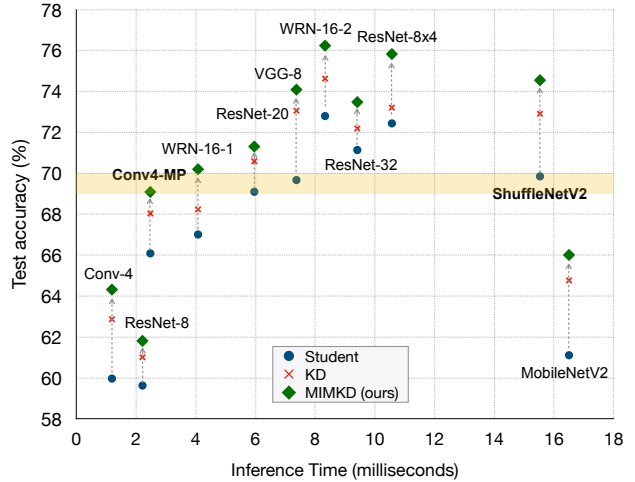


Figure 1. Accuracy-efficiency trade-off of various CNN models on the CIFAR100 dataset. Inference time is the average over a single input. MIMKD provides significant gains over baseline student network accuracies and KD [12] across a range of student-teacher pairs. Notice that Conv4-MP, which is a simple 4-layer CNN, after distillation performs close to a more standard ShuffleNetV2. *Note:* Runtimes were computed on an Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz.

In this paper, we look at knowledge distillation from an information-theoretic perspective. For better distillation, the student needs to generate representations that share maximum information with the representations generated by the teacher. Based on this intuition, we propose Mutual Information Maximization Knowledge Distillation (MIMKD). Multiple approaches have been proposed to estimate the mutual information between high-dimensional continuous variables [2, 13]. Belghazi et al [2] propose a KL-divergence based formulation of mutual information. We observe that this approach can be extended to maximize the mutual information in a contrastive setup. Contrastive methods have had an outsized impact in other problems such as self-supervised learning [6, 10], however they rely on sampling a rather large number of paired inputs to optimize their objective functions. We find that by using a Jensen-Shannon divergence (JSD) based formulation we obtain a more stable objective to optimize where the perfor-

mance is invariant to the number of negative samples while being monotonically related to the true mutual information as also shown in Hjelm et al [13].

The previous work of Hinton *et al.* [12] performed distillation by minimizing the Kullback–Leibler divergence between the output logits of two models. Since then, several other output-based knowledge distillation approaches have been developed [5, 12, 25, 32]. Some of these methods try to match the final outputs of two networks by minimizing a distance metric. Other works have also encouraged additional knowledge transfer by minimizing a metric between intermediate representations [1, 20, 29]. However, models with significantly different architectures have distinct data-abstraction flows, and the complexity of the patterns recognized at different depths in the model varies significantly with model architecture (i.e. the number of filters in convolutional layers). Therefore, minimizing a non-parameterized distance metric on the representations imposes an additional structural constraint that might not be ideal for knowledge transfer. In our work, we are still able to leverage both local and global information by maximizing mutual information instead of a rigid distance metric between representations.

More comparable to our work is the recently proposed Contrastive Representation Distillation (CRD) framework [25]. This method uses a Noise Contrastive Estimation (NCE) objective [9, 17] to transfer structured relational knowledge from the teacher to the student. However, a caveat of this approach is that it requires comparing features from a large number of images simultaneously. In practice, this requires large batches [6] or memory banks [10, 27]. In contrast, our JSD-based contrastive objective uses only one negative sample as opposed to the 16,384 samples used in CRD [25]. This enables us to impose additional region-consistent local and feature-level constraints as large number of negative samples for intermediate features are not required.

We propose three mutual information maximization objectives between the teacher and student networks: (1) *Global information maximization*, which aims to maximize the shared information between the final output representations. This pushes the student network to generate feature vectors that are as rich as the ones generated by the teacher. (2) *Local information maximization*, which pushes the student network to recognize complex patterns from each region of the image that are ultimately useful for classification. This is achieved by maximizing the mutual information between region-specific vectors extracted from an intermediate representation of the student network and the final representation of the teacher network. Finally, (3) *Feature Information Maximization*, which is designed to structurally improve the granular feature-extraction capability of the student by maximizing the mutual information between

region-consistent local vectors extracted from intermediate representations of the networks.

Our experimental results in Section 4 demonstrate that these objectives are more effective than competing methods across a wide range of student-teacher pairs and conduct extensive ablation studies of the effect of each proposed objective. We particularly demonstrate the effectiveness of our method in knowledge distillation across student-teacher network pairs with different capacities, student-teacher network pairs with different architectures, and in the extreme case where student networks are extremely low capacity. We show that MIMKD provides consistently better results across all these testing scenarios. Figure 1 shows a summary of results for various student networks on CIFAR-100 when compared to regular KLD-based knowledge distillation [12]. Moreover, we compare the transferability of features learned with knowledge distillation from MIMKD and SOTA baseline. Our results show that MIMKD learns more general and transferrable representations.

## 2. Related Work

In this section, we discuss previous efforts in improving knowledge distillation, and in estimating mutual information which are the key areas of contribution of our work.

### 2.1. Knowledge distillation.

The concept of knowledge distillation (KD) was introduced in the works of Buciluă *et al.* [4] and later formalized for deep neural networks by Hinton *et al.* [12]. In knowledge distillation, the goal is to train smaller models that can mimic the performance of larger models. Hinton *et al.* [12] proposed a knowledge distillation method in which the student network is trained using soft labels extracted from teacher networks.

Attention transfer [29] introduced the idea of transferring intermediate attention maps from the teacher to the student network. Fitnets [20] also presented the idea of adding more supervision by matching the intermediate representation using regressors. Yim *et al.* [28] formulated the distillation problem using the flow of solution procedure (FSP), which is computed as the gram matrix of features across layers. Sau *et al.* [23] proposed to include a noise-based regularizer while training the student with the teacher. Specifically, they perform perturbation in the logits of the teacher as a regularization approach. In Correlation Congruence for Knowledge Distillation (CCKD) [19], the authors present a framework which transfers not only instance-level information but also the correlation between instances. In CCKD, a Taylor series expansion-based kernel method is proposed to better capture the correlation between instances. Tung *et al.* [26] propose a loss that is based on the observation that semantically similar inputs produce similar activation patterns in trained networks. Variational Information Distilla-

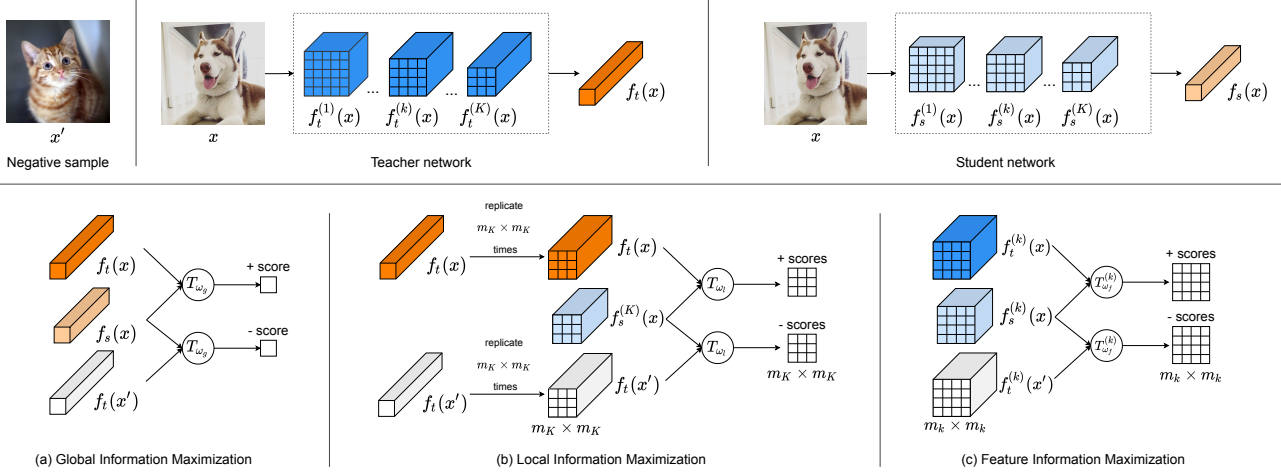


Figure 2. Overall schematic of our proposed method for mutual information maximization based knowledge distillation (MIMKD). **Top:** Representations generated by teacher and student networks for image  $x$  and a negative sample  $x'$ . Note that our method uses only one negative sample. **Bottom:** (a) Positive and negative pairs of final feature vectors are passed into the discriminator function to get scores. (b) Teacher’s final representation is replicated to match student’s last intermediate representation. (c) For each group of same-sized intermediate feature maps in set  $\mathcal{R}$ , positive and negative pairs are passed into a distinct discriminator function to get scores. The positive and negative scores obtained are then used with equation (2) to estimate and maximize a lower-bound on mutual information.

tion (VID) [1] uses a variational lower-bound for the mutual information between the teacher and the student representations by approximating an intractable conditional distribution using a pre-defined variational distribution.

More closely related to our work are recently proposed methods that cast knowledge distillation as a mutual information maximization problem. Contrastive representation distillation (CRD) [25] used a contrastive objective similar to Oord *et al.* [17] to maximize a lower-bound on mutual information between final representations. The objective used by CRD is a strong lower-bound on the mutual information but requires a significant number of negative samples during training, consequently, requiring large batch-sizes or memory buffers. These practical constraints become even more limiting if mutual information needs to be minimized at the feature-level to enforce regional-supervision during student training. Our work proposes an alternative that bypasses the needed for such large batch-sizes and thus enables to optimize for mutual information through three separate objectives.

## 2.2. Mutual Information Estimation.

Mutual information is a fundamental quantity that measures the relationship between random variables but it is notoriously difficult to measure [18]. An exact estimate is only tractable for discrete variables or a small set of problems where the probability distributions are known. However, both the mentioned scenarios are unlikely for real-world visual datasets. Recently, Mutual Information Neural Estimation (MINE) [2] demonstrated a strong method for estimation

of mutual information between high-dimensional continuous random variables using neural networks and gradient descent. MINE [2] proposed a general-purpose parametric neural estimator of mutual information based on dual representations of the KL-divergence [21]. Following from MINE [2], Deep InfoMax [13] proposed a mutual information based objective for unsupervised representation learning. Deep InfoMax [13] contends that it is unnecessary to use the exact KL-divergence based formulation of mutual information and demonstrated the use of an alternative formulation based on the Jensen-Shannon divergence (JSD). The authors showed that the JSD based estimator is stable, and does not require a large number of negative samples. In addition, Deep InfoMax [13] also demonstrated the value of including global and local structure-based mutual information objectives for representation learning. We leverage this line of work in our method to propose a framework for knowledge distillation that leverages both local and global features without significantly adding memory overheads during training.

## 3. Method

In this section, we describe our general framework for model compression or knowledge distillation in a teacher student setup. Consider a stronger teacher network  $f_t : X \rightarrow Y$  with trained parameters  $\phi$  and a student network, operating on the same domain,  $f_s : X \rightarrow Y$  with parameters  $\theta$ . Let  $x$  be the sample drawn from the data distribution  $p(x)$  and  $f_t(x)$  &  $f_s(x)$  denote the representations extracted

from the pre-classification layer, while  $f_t^{cls}(x)$  &  $f_s^{cls}(x)$  denote the predicted class-probability distributions from the teacher and the student networks respectively. Now consider a set  $\mathcal{R} = \{(f_t^{(k)}(x), f_s^{(k)}(x))\}_{k=1}^K$  that contains  $K$  pairs of intermediate representations extracted from the networks such that each pair in set  $\mathcal{R}$  contains same-sized intermediate representations extracted from the networks, where  $m_k \times m_k$  is the size corresponding to the  $k$ -th pair in the set. Each location in these 2-dimensional intermediate representations corresponds to a specific region in the input image. Note that we do not include the final representations  $f_t(x)$  and  $f_s(x)$  in the set  $\mathcal{R}$ .

Our method focuses on maximizing the mutual information, (1) between final image representations  $f_t(x)$  and  $f_s(x)$  (global information maximization), (2) between the global image representation from the teacher network  $f_t(x)$  and the last intermediate representation from the student network  $f_s^{(K)}(x)$  (local information maximization), and (3) between the pairs in set  $\mathcal{R}$  (feature information maximization). Figure 2 shows an overview of our method.

### 3.1. Mutual Information Maximization

In order to estimate and maximize mutual information between random variables  $X$  and  $Z$ , we train a neural network to distinguish samples generated from the joint distribution,  $P(X, Z)$  and the product of marginals  $P(X)P(Z)$ . In MINE [2], the authors use the Donsker-Varadhan (DV) [8] representation of the KL-divergence as the lower bound on the mutual information,

$$\begin{aligned} I(X; Z) &:= D_{KL}(P(X, Z) \parallel P(X)P(Z)) \\ &\geq \hat{I}_\omega^{DV}(X; Z) := \mathbb{E}_{P(X, Z)}[T_\omega] \\ &\quad - \log \mathbb{E}_{P(X)P(Z)}[e^{T_\omega}], \quad (1) \end{aligned}$$

where  $T_\omega : \mathcal{X} \times \mathcal{Z} \rightarrow \mathbb{R}$  is the discriminator neural network with parameters  $\omega$ .

Another popular bound on mutual information is formulated as infoNCE [17] based on Noise-Contrastive Estimation [9]. However, as demonstrated in [13], both DV and infoNCE require a large number of negative samples during training. Recent works tackle this problem by using a memory-buffer that keeps representations from previous samples in memory to be accessed during training. As implemented in CRD [25], this can be done if mutual information is maximized only between the final representations of the networks as the dimensions of the representations to be kept in memory is limited. In our setup, we require negative samples for each location in the multiple  $K$  intermediate feature maps as well as for the final representations. This becomes unfeasible for most large state-of-the-art architectures.

Different from above methods, in our approach we use Jensen-Shannon divergence based mutual information estimation, similar to the formulations in [16] and [3],

$$\begin{aligned} I(X; Z) &\geq \hat{I}_\omega^{JSD}(X; Z) := \mathbb{E}_{P(X, Z)}[-\log(1 + e^{-T_\omega})] \\ &\quad - \mathbb{E}_{P(X)P(Z)}[\log(1 + e^{T_\omega})]. \quad (2) \end{aligned}$$

Overall, we optimize the parameters  $\theta$  of the student network  $f_s$  and parameters  $\omega$  of the critic network  $T_\omega$  by simultaneously estimating and maximizing mutual information between the representations of the frozen teacher network and the student network.

### 3.2. Global information maximization

Our global objective aims to maximize the mutual information between the richer final representation of the frozen teacher network  $f_t(x)$  and the final representation of the student network  $f_s(x)$  to encourage the student to learn richer representations. This objective uses a discriminator function  $T_{\omega_g}$ , where  $\omega_g$  are the trainable parameters. We optimize the parameters of the student and the discriminator function simultaneously as:

$$(\hat{\omega}_g, \hat{\theta}) = \operatorname{argmax}_{\omega_g, \theta} \hat{I}_{\omega_g}(f_t(x), f_s(x)). \quad (3)$$

### 3.3. Local information maximization

In this objective we maximize the mutual information between a richer final representation of the teacher network and representations of local regions extracted by the student network. This objective draws from the assertion that the final teacher representations contains valuable information required for downstream classification. Hence, this objective encourages the student network to extract information from local image regions that is ultimately useful for classification.

We enforce this objective between  $f_t(x)$  and the last intermediate representation from the student network in the set  $\mathcal{R}$ . Therefore for  $k = K$ ,  $f_s^{(K)}(x)$  is a  $m_K \times m_K$  feature map where each location roughly corresponds to an  $H/m_K \times W/m_K$  patch in the input image where  $H, W$  are the height and width of the image. The representation of each such patch  $\{f_s^{(K)}(x)\}_{i,j}$  is then paired with  $f_t(x)$ , where  $i, j \in [1, m_K]$  denotes the specific location in the feature map. The pairs are then used with the mutual information estimator to optimize the parameters as follows:

$$(\hat{\omega}_l, \hat{\theta}) = \operatorname{argmax}_{\omega_l, \theta} \frac{1}{m_K^2} \sum_{i=1}^{m_K} \sum_{j=1}^{m_K} \hat{I}_{\omega_l}(f_t(x), \{f_s^{(K)}(x)\}_{i,j}), \quad (4)$$

where a discriminator neural network  $T_{\omega_l}$  with parameters  $\omega_l$  is used.

Student Network	WRN-16-1	WRN-16-2	ResNet-8	ResNet-20	ResNet-20	ResNet-8x4	VGG-8
Teacher Network	WRN-40-2	WRN-40-2	ResNet-110	ResNet-110	ResNet-56	ResNet-32x4	VGG-19
Student Accuracy	67.01	72.80	59.63	69.10	69.10	72.44	69.67
Teacher Accuracy	75.31 <sub>+8.30</sub>	75.31 <sub>+2.51</sub>	73.82 <sub>+14.19</sub>	73.82 <sub>+4.72</sub>	72.31 <sub>+3.21</sub>	79.24 <sub>+3.80</sub>	74.63 <sub>+4.96</sub>
FitNets [20]	68.35 <sub>+1.34</sub>	73.11 <sub>+0.31</sub>	60.36 <sub>+0.73</sub>	69.12 <sub>+0.02</sub>	69.28 <sub>+0.18</sub>	73.80 <sub>+1.36</sub>	71.32 <sub>+1.65</sub>
Attention Transfer (AT) [29]	68.49 <sub>+1.48</sub>	73.37 <sub>+0.57</sub>	60.24 <sub>+0.61</sub>	70.36 <sub>+1.26</sub>	70.18 <sub>+1.08</sub>	73.20 <sub>+0.76</sub>	71.71 <sub>+2.04</sub>
Variational Inf. Distill. (VID) [1]	68.95 <sub>+1.94</sub>	73.89 <sub>+1.09</sub>	60.44 <sub>+0.81</sub>	70.32 <sub>+1.22</sub>	70.52 <sub>+1.42</sub>	73.19 <sub>+0.75</sub>	71.52 <sub>+1.85</sub>
Knowledge Distillation (KD) [12]	68.24 <sub>+1.23</sub>	73.91 <sub>+1.11</sub>	61.01 <sub>+1.38</sub>	70.32 <sub>+1.22</sub>	70.59 <sub>+1.40</sub>	73.21 <sub>+0.77</sub>	72.29 <sub>+2.62</sub>
Contrastive Repr. Distill (CRD) [25]	69.21 <sub>+2.20</sub>	74.17 <sub>+1.37</sub>	60.82 <sub>+1.19</sub>	<b>71.45</b> <sub>+2.35</sub>	71.12 <sub>+2.02</sub>	75.21 <sub>+2.77</sub>	73.10 <sub>+3.43</sub>
MIMKD (this work)	<b>70.20</b> <sub>+3.19</sub>	<b>75.16</b> <sub>+2.36</sub>	<b>61.81</b> <sub>+2.18</sub>	71.43 <sub>+2.33</sub>	<b>71.31</b> <sub>+2.21</sub>	<b>75.83</b> <sub>+3.39</sub>	<b>73.27</b> <sub>+3.60</sub>

Table 1. Observed test accuracy (in %) of student networks trained with teacher networks of higher capacity but similar architecture on the CIFAR100 dataset using MIMKD and other competing methods. MIMKD shows consistent increases in accuracy for all model pairs and the largest gains overall.

### 3.4. Feature Information maximization

This objective aims to maximize the mutual information between region-consistent intermediate representations from the networks. In neural networks, the complexity of captured visual patterns increases towards the later layers [31]. Intuitively, to mimic the representational power of the teacher, the student network needs to learn these complex patterns hierarchically. In order to motivate such hierarchical learning, mutual information is maximized between intermediate features at different depths in the networks. This enables the student to learn to identify complex patterns in a bottom-up fashion and systematically learn to generate richer features. Note that within each pair of intermediate feature maps in set  $\mathcal{R}$ , mutual information is maximized between vectors corresponding to the same location in the image. This information maximization pushes the student network to extract features from each region of the image that share maximum information with the features extracted by the teacher network from the same region. For a pair  $(f_t^{(k)}(x), f_s^{(k)}(x)) \in \mathcal{R}$ , information is maximized between pairs of region-consistent vectors  $\{f_t^{(k)}(x)\}_{i,j}$  and  $\{f_s^{(k)}(x)\}_{i,j}$  for each  $i, j \in [1, m_k]$  as follows:

$$(\hat{\omega}_f, \hat{\theta}) = \arg\max_{\omega_f, \theta} \frac{1}{K} \frac{1}{m_k^2} \sum_{k=1}^K \sum_{i=1}^{m_k} \sum_{j=1}^{m_k} \hat{I}_{\omega_f}(\{f_t^{(k)}(x)\}_{i,j}, \{f_s^{(k)}(x)\}_{i,j}),$$

where a discriminator neural network  $T_{\omega_f}$  with parameters  $\omega_f$  is used.

### 3.5. Divergence objective

This objective aims to minimize the divergence between the predicted class-probability distributions from teacher and student models. In order to do this, KD [12] minimizes the Kullback-Leibler divergence (KLD). However,

KL-divergence serves as poor distance-metric due to its non-symmetric nature. Therefore, we formulate an objective based on the Jensen-Shannon Divergence (JSD) to measure the similarity between probability distributions. Overall, due to the favorable properties of JSD over KLD, our objective is a symmetric and smoothed version of the its KLD-based counterpart and can be used as a distance metric.

We minimize the divergence between the output probability distributions  $f_t^{cls}(x)$  and  $f_s^{cls}(x)$ , from the teacher and the student, using the following objective:

$$(\hat{\theta}) = \arg\min_{\theta} \frac{KL(f_t^{cls}(x) \parallel Q)}{2} + \frac{KL(f_s^{cls}(x) \parallel Q)}{2},$$

where,

$$Q = \frac{f_t^{cls}(x) + f_s^{cls}(x)}{2}. \quad (5)$$

This objective functions as label-smoothing during training by encouraging the student to mimic the softened class scores of the teacher.

### 3.6. Classification objective

Here the cross-entropy loss is minimized between the output of the classification function  $f_s^{cls}(x)$  and the target label  $y$  as follows:

$$(\hat{\theta}) = \arg\min_{\theta} \mathcal{L}_{CE}(y, f_s^{cls}(x)), \quad (6)$$

where  $\mathcal{L}_{CE}$  denotes the cross-entropy function.

Our overall objective is a weighted-summation of all the above individual objectives with weights  $\alpha$  (cross-entropy loss),  $1 - \alpha$  (JS-divergence objective),  $\lambda_g$  (global MI maximization),  $\lambda_l$  (local MI maximization), and  $\lambda_f$  (feature MI maximization)



Student Network	WRN-16-1	WRN-16-2	VGG-8	ShuffleNetV1	ShuffleNetV2	MobileNetV2
Teacher Network	ResNet-110	ResNet-32x4	ResNet-32x4	VGG-13	ResNet-50	VGG-13
Student Accuracy	67.01	72.80	69.67	70.51	69.85	61.11
Teacher Accuracy	73.82 <sub>+6.81</sub>	79.24 <sub>+6.44</sub>	79.24 <sub>+9.57</sub>	74.62 <sub>+4.11</sub>	79.23 <sub>+9.38</sub>	74.62 <sub>+13.51</sub>
FitNets [20]	67.99 <sub>+0.98</sub>	73.79 <sub>+0.99</sub>	70.28 <sub>+0.61</sub>	72.29 <sub>+1.78</sub>	71.80 <sub>+1.95</sub>	61.42 <sub>+0.31</sub>
Attention Transfer (AT) [29]	66.42 <sub>-0.59</sub>	72.19 <sub>-0.61</sub>	71.77 <sub>+2.10</sub>	71.19 <sub>+0.68</sub>	70.78 <sub>+0.93</sub>	61.96 <sub>+0.85</sub>
Variational Inf. Distill. (VID) [1]	67.47 <sub>+0.46</sub>	73.38 <sub>+0.58</sub>	71.52 <sub>+1.85</sub>	72.22 <sub>+1.71</sub>	72.84 <sub>+2.99</sub>	63.01 <sub>+1.90</sub>
Knowledge Distillation (KD) [12]	68.86 <sub>+1.85</sub>	74.63 <sub>+1.83</sub>	73.46 <sub>+3.79</sub>	72.26 <sub>+1.75</sub>	72.91 <sub>+3.06</sub>	64.47 <sub>+3.36</sub>
Contrastive Repr. Distill. (CRD) [25]	69.71 <sub>+2.70</sub>	75.61 <sub>+2.81</sub>	73.73 <sub>+4.06</sub>	72.86 <sub>+2.35</sub>	73.65 <sub>+3.80</sub>	<b>66.34<sub>+5.23</sub></b>
MIMKD (this work)	<b>69.88<sub>+2.87</sub></b>	<b>76.24<sub>+3.44</sub></b>	<b>74.09<sub>+4.42</sub></b>	<b>73.88<sub>+3.37</sub></b>	<b>74.55<sub>+4.70</sub></b>	65.89 <sub>+4.78</sub>

Table 2. Observed test accuracy (in %) of student networks trained with teacher networks of higher capacity and different architecture on the CIFAR100 dataset using our method MIMKD and other distillation frameworks.

## 4. Experiments

We evaluate the performance of our approach on the task of model compression on the CIFAR-100 dataset [15] in two distinct setups. First, we perform model compression between pairs of (1) similar standard CNN architectures of varying capacities and (2) between pairs of significantly different standard CNN architectures. Second, we perform model compression via knowledge distillation from standard CNN architectures to shallower custom-designed CNN architectures. The CIFAR-100 dataset contains colored natural images of size  $32 \times 32$ . It has 50K training images with 500 images in each of 100 classes and a total of 10K test images. In our experiments, we use standard CNN architectures of varied capacities, such as ResNet [11], Wide ResNet (WRN) [30], MobileNet [22], ShuffleNet [33], and VGG [24].

We compare our method with other knowledge distillation methods, such as (1) Knowledge Distillation (KD) [12], (2) FitNets [20], (3) Attention Transfer (AT) [29], (4) Variational Information Distillation (VID) [1], and (5) Contrastive Representation Distillation (CRD) [25]. We used the following values for hyper-parameters based on a held out set:  $\alpha = 0.9$ ,  $\lambda_g = 0.2$ ,  $\lambda_l = 0.8$ ,  $\lambda_f = 0.8$  for all our experiments. The hyper-parameter choice for other approaches can be found in the appendix.

Additionally, in order to demonstrate the scalability of our method, we compare our distillation performance on the ImageNet [7] dataset against AT [29], and KD [12]. ImageNet is a large-scale dataset with 1.2 million training images across 1K classes and a total of 50K validation images.

Finally, we compare MIMKD with competing approaches on their ability to learn transferrable representations.

### 4.1. Implementation Details

We adopted the generally established approach for training CNNs on the CIFAR-100 dataset. We use SGD with momentum 0.9, weight decay  $5 \times 10^{-4}$ , and an initial learning rate of 0.05 for a total of 240 epochs with batch-size 64. The learning rate is decayed by 0.1 at the 150th, 180th and the 210th epoch. We used random horizontal flips and random crop for augmenting the dataset during training. For ImageNet, we use the standard PyTorch training scheme for ResNets [11]. The parameterized mutual information discriminator functions ( $T_{\omega_g}$ ,  $T_{\omega_l}$ , &  $T_{\omega_f}$ ) can be modeled by arbitrary neural networks. In our experiments, we use two distinct architectures described in detail in the appendix A.3. We will release our source code to reproduce experiments in our paper upon publication.

## 5. Results

In this section we demonstrate the prowess of our distillation framework based on model compression performance in the following setups: Under similar student-teacher network architectures (sec. 5.1), under dissimilar architectures (sec. 5.2), under a setting with custom designed shallow student networks (sec. 5.3), in a larger scale setting on ImageNet (sec. 5.4), and in terms of transfer learning performance (sec. 5.5).

### 5.1. Similar CNN Architectures.

We perform knowledge distillation from a teacher network to a student network of the same family (e.g. ResNets of different capacities). Table 1 presents our results, showing that our method outperforms others in most setups and always obtains gains with respect to student networks. Notice that CRD [25] is able to slightly surpass the performance of our method in one setup while being close in most cases. We find this encouraging as CRD [25] uses a similar mutual information maximization based formu-

Student Network	Conv-4			Conv-4-MP		
	ResNet-110	VGG-13	ResNet-32x4	ResNet-110	VGG-13	ResNet-32x4
Teacher Accuracy	59.97	59.97	59.97	66.09	66.09	66.09
Student Accuracy	73.82 <sub>+13.85</sub>	74.62 <sub>+14.65</sub>	79.24 <sub>+19.27</sub>	73.82 <sub>+7.73</sub>	74.62 <sub>+8.53</sub>	79.24 <sub>+13.15</sub>
FitNets [20]	60.58 <sub>+0.61</sub>	61.81 <sub>+1.84</sub>	62.89 <sub>+2.92</sub>	67.38 <sub>+1.29</sub>	66.52 <sub>+0.43</sub>	67.21 <sub>+1.12</sub>
Attention Transfer (AT) [29]	61.65 <sub>+1.68</sub>	62.16 <sub>+2.19</sub>	63.10 <sub>+3.13</sub>	67.52 <sub>+1.43</sub>	66.21 <sub>+0.12</sub>	66.03 <sub>-0.06</sub>
Variational Inf. Distill. (VID) [1]	61.93 <sub>+1.96</sub>	62.49 <sub>+2.52</sub>	63.45 <sub>+3.48</sub>	67.76 <sub>+1.67</sub>	67.40 <sub>+1.31</sub>	67.86 <sub>+1.77</sub>
Knowledge Distillation (KD) [12]	61.98 <sub>+2.01</sub>	62.10 <sub>+2.13</sub>	62.87 <sub>+2.90</sub>	67.51 <sub>+1.42</sub>	67.84 <sub>+1.75</sub>	68.04 <sub>+1.95</sub>
Contrastive Repr. Distill. (CRD) [25]	62.13 <sub>+2.16</sub>	62.54 <sub>+2.57</sub>	63.76 <sub>+3.79</sub>	67.96 <sub>+1.87</sub>	68.06 <sub>+1.97</sub>	68.52 <sub>+2.43</sub>
MIMKD (this work)	<b>62.91<sub>+2.94</sub></b>	<b>62.95<sub>+2.98</sub></b>	<b>64.32<sub>+4.35</sub></b>	<b>68.77<sub>+2.68</sub></b>	<b>68.91<sub>+2.82</sub></b>	<b>69.09<sub>+3.00</sub></b>

Table 3. Observed test accuracy (in %) of shallow student networks trained with teacher networks of higher capacity and standard architectures on the CIFAR100 dataset using our methods MIMKD and other distillation frameworks.

lation in their distillation objective with a tighter lower-bound. Therefore, if we only use the global objective in our method, CRD [25] should outperform our method due to its tighter bound. Despite compromising the lower bound on mutual information, MIMKD takes advantage of using region-consistent local and feature-level mutual information maximization.

## 5.2. Dissimilar CNN Architectures.

Here, we perform knowledge distillation from a teacher network to a student network with a significantly different architecture. This tests the flexibility methods to adapt to distinct data-abstraction flows of dissimilar neural network architectures. Table 2 demonstrates that our method (MIMKD) outperforms other distillation methods in most teacher-student combinations increasing the accuracy of a ShuffleNetV2 by 4.7% while distilling from a much different ResNet-50 model. This demonstrates that our method is able to accommodate significant architectural differences in teacher-student pairs and does not impose structural constraints on intermediate layers that hinder training. While other methods that work on intermediate feature maps like AT [29] and FitNets [20] do not show much improvement from base student accuracy.

## 5.3. Shallow CNN Architectures.

In this section, we describe our experiments where we distill knowledge from a standard teacher network into a shallow custom-designed CNN. This is done to demonstrate that it is feasible to design and distill information into light-weight models such that they perform competitively with standard CNN architectures while running faster. For our experiments we use 2 shallow CNNs; (1) Conv-4 with 4 convolutional-blocks followed by average pooling operation and a linear layer, where each convolutional-block is made-up of a convolutional layer with kernel size  $3 \times 3$  and

stride 2 followed by batch-normalization and a ReLU non-linearity, (2) Conv-4-MP which has 4 convolutions blocks followed by average pooling and a linear layer at the end, where each convolutional-block contains a convolutional layer with kernel size  $3 \times 3$  and stride 1 followed by batch-normalization, ReLU and a max-pooling layer. These architectures were chosen as they are compact and run relatively faster on standard CPUs (see Figure 1). Table 3 compiles our results compared to other distillation methods for custom-designed shallow CNN architectures. Notice how a simple model such as Conv-4-MP becomes competitive with ShuffleNetV2’s base student accuracy. Our method is able to outperform all other methods in this setup. Additionally, we can see that distillation is most successful with ResNet-32x4 as the teacher than for other architectures. This could be because of the larger gap in the baseline accuracy of the networks. Under this more controlled experiment with fixed students, larger gaps between student-teacher pairs also led to larger gains after distillation.

## 5.4. ImageNet results.

In this experiment we train a student ResNet-18 with a pre-trained teacher ResNet-34 on the ImageNet dataset (ILSVRC). Note that we do not perform any hyper-parameter tuning specifically for this configuration and use the same values we obtained for the CIFAR-100 dataset i.e.  $\alpha = 0.9$ ,  $\lambda_g = 0.2$ ,  $\lambda_l = 0.8$ ,  $\lambda_f = 0.8$ . We observed that our method is able to reduce the gap between the teacher and the student performance by 1.44%. Results are presented in Table 4.

## 5.5. Transferring representations

Finally, we compare the transferability of the features learned with knowledge distillation from MIMKD and baselines, on two other datasets: STL-10 and TinyImageNet. A WRN-16-2 network is trained with and without

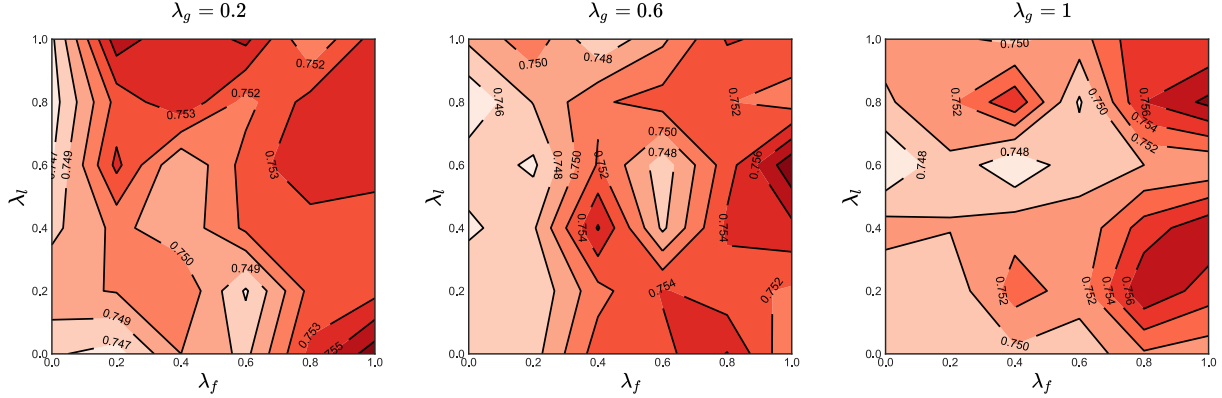


Figure 3. Results from the ablation studies on CIFAR100 dataset using a student ResNet-8x4 (baseline acc. 72.44%) with teacher ResNet-32x4 (baseline acc. 79.24%). Contour lines represent the final test accuracy of the student. Grid search was performed by varying the values of  $\lambda_f$ ,  $\lambda_g$ ,  $\lambda_l$  from 0 to 1 with increments of 0.2 while  $\alpha$  was kept constant at 1. In each plot, the accuracy landscape is shown with  $\lambda_g$  set to a constant value. Plots for remaining values of  $\lambda_g$  have been added to the appendix.

Student Network	ResNet-18
Teacher Network	ResNet-34
Student Accuracy	68.88
Teacher Accuracy	72.82 <sup>+3.94</sup>
Knowledge Distill. (KD) [12]	69.66 <sup>+0.78</sup>
Attention Transfer (AT) [29]	69.70 <sup>+0.82</sup>
MIMKD (this work)	<b>70.32<sup>+1.44</sup></b>

Table 4. Observed top-1 validation accuracy (in %) of the student network on the ImageNet dataset using our method (MIMKD) and other distillation frameworks. In similar settings, the more recent Contrastive Representation Distillation (CRD) method reports comparable performance with an improvement of +1.42 from a student network [25].

distillation from a pre-trained WRN-40-2 teacher on the CIFAR100 dataset. The student is then used as a frozen feature extractor (pre-classification layer) for images in the STL-10 and the TinyImageNet dataset. A linear classifier is trained on these extracted features to perform classification on the test sets of these datasets. The classification accuracy on the unseen datasets is interpreted as the transferability of representations. Results are presented in Table 5 and show that MIMKD learns more transferrable representations.

	STL-10	TinyImageNet
Base Accuracy (no distillation)	69.5	33.8
Knowledge Distillation (KD)	70.6	33.9
Attention Transfer (AT)	70.8	34.4
Contrastive Repr. Distill (CRD)	71.4	35.6
MIMKD (this work)	<b>71.8</b>	<b>36.2</b>

Table 5. Observed test-set accuracy (in %) of the student network on STL-10 and TinyImagenet datasets using our method (MIMKD) and other distillation frameworks.

## 5.6. Ablation Study

We perform an extensive ablation study to demonstrate the value of each component of our mutual information maximization objective. Ablative study experiments are performed with ResNet-32x4 as the teacher network and ResNet-8x4 as the student network where the baseline accuracy of the teacher is 79.24% and that of the student network is 72.44%. The values of the hyper-parameters  $\lambda_g$ ,  $\lambda_l$  and  $\lambda_f$  — that control the weight of the global, local and feature mutual information maximization objectives respectively — were varied between 0 and 1 with an increment of 0.2. The contour plots in Figure 3 shows the test accuracy landscape with respect to the values of the hyper-parameters. It can be seen from the figure that for any value of  $\lambda_g$ , we can observe generally higher performance towards higher values of  $\lambda_f$ . This shows the value of maximizing region-consistent mutual information between representations. Please refer to the appendix for additional accuracy landscape plots.

## 6. Conclusion

In this paper, we presented a novel Mutual Information Maximization based knowledge distillation framework (MIMKD). Our method was formulated using the JSD based lower-bound on mutual information which we optimize using only one negative sample. We presented three information maximization formulations and demonstrated the value of region-consistent information maximization on distillation performance. Further works could explore our contention that if used with a tighter lower-bound, our information maximization objectives have the potential to surpass even its current performance. The role and the architecture of the discriminator functions can also be explored further in future works.



## 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 9163–9171, 2019. 2, 3, 5, 6, 7, 11
- [2] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International Conference on Machine Learning*, pages 531–540. PMLR, 2018. 1, 3, 4
- [3] Philemon Brakel and Yoshua Bengio. Learning independent features with adversarial nets for non-linear ica. *arXiv preprint arXiv:1710.05050*, 2017. 4
- [4] Cristian Bucilua, 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. 1, 2
- [5] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 742–751, 2017. 2
- [6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 2
- [7] 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. 6
- [8] Monroe D Donsker and SR Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212, 1983. 4
- [9] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010. 2, 4
- [10] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 1, 2
- [11] 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*, pages 770–778, 2016. 6
- [12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1, 2, 5, 6, 7, 8, 11
- [13] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018. 1, 2, 3, 4, 11
- [14] Zehao Huang and Naiyan Wang. Data-driven sparse structure selection for deep neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 304–320, 2018. 1
- [15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6
- [16] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. *arXiv preprint arXiv:1606.00709*, 2016. 4
- [17] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 3, 4
- [18] Liam Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003. 3
- [19] 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/CVF International Conference on Computer Vision*, pages 5007–5016, 2019. 2
- [20] 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. 2, 5, 6, 7, 11
- [21] Avraham Ruderman, Mark Reid, Darío García-García, and James Petterson. Tighter variational representations of f-divergences via restriction to probability measures. *arXiv preprint arXiv:1206.4664*, 2012. 3
- [22] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 6
- [23] Bharat Bhushan Sau and Vineeth N Balasubramanian. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*, 2016. 2
- [24] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6
- [25] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019. 2, 3, 4, 5, 6, 7, 8, 11
- [26] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1365–1374, 2019. 2
- [27] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018. 2
- [28] 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. 2

- [29] 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. 2, 5, 6, 7, 8, 11
- [30] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 6
- [31] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014. 5
- [32] Feng Zhang, Xiatian Zhu, and Mao Ye. Fast human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3517–3526, 2019. 2
- [33] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018. 6

## A. Appendix

### A.1. Limitations and Broader Impacts

In this paper, we presented a novel Mutual Information Maximization based knowledge distillation framework (MIMKD). Our method uses the JSD based lower-bound on mutual information which is optimized using only one negative sample. However, despite its favorable properties, our lower-bound may be less tight on the mutual information than the infoNCE bound as it approximates the mutual information by being monotonically related with it. Additionally, as we use only one negative sample, the performance of the method may be hindered by the presence of false negatives. The performance of the method is also effected by the architecture of the discriminator functions which can be explored further. We presented three information maximization formulations and demonstrated the value of region-consistent information maximization on distillation performance. We observe that the performance is slightly-sensitive to the hyper-parameters that control the relative value of our global, local, and feature information maximization formulations. This has been explored in great detail in our ablation sections and further demonstrated in figures 4, 5, and 6. Our method transfers representations from the teacher to the student. As such, harmful biases that the teacher has learnt are transferred to the student as well. And further exploration is required to alleviate the transfer of such biases during distillation.

### A.2. Hyper-parameters for other methods

The student is trained with the following loss function which is a combination of the distillation loss and the cross-entropy loss for classification:

$$\mathcal{L} = \alpha \mathcal{L}_{cls} + (1 - \alpha) \mathcal{L}_{KD} + \beta \mathcal{L}_{dis} \quad (7)$$

Note that we set  $\alpha = 1$  for all methods except KD [12] and the value of  $\beta$  is set to the value recommended in the original work as follows:

1. KD [12]:  $\alpha = 0.9, \beta = 0$
2. Fitnet [20]:  $\beta = 100$
3. AT [29]:  $\beta = 1000$
4. VID [1]:  $\beta = 1$
5. CRD [25]:  $\beta = 0.8$ , for CRD evaluation, we use a original work inspired self-implementation with 1024 negative samples and  $i \neq j$  negative sampling methodology as described in the original work.

### A.3. Pairing Intermediate Representations

#### A.3.1 Similar CNN Architectures.

Consider the case of distillation when the teacher network is a pre-trained WRN-40-2 and the student network is a WRN-16-1. We use 4 same-sized representations extracted from intermediate layers of the networks. Therefore, the set  $\mathcal{R} = \{(f_t^{(k)}(x), f_s^{(k)}(x))\}_{k=1}^K$  contains  $k$  pairs of same-sized 2-dimensional representations. Table 6 describes the sizes of the intermediate representations used for feature-based mutual information maximization. It can be seen that for this combination we use  $k = 4$  in our formulation.

	WRN-40-2	WRN-16-1
k	$f_t^{(k)}(x)$	$f_s^{(k)}(x)$
1	$16 \times 32 \times 32$	$16 \times 32 \times 32$
2	$32 \times 32 \times 32$	$16 \times 32 \times 32$
3	$64 \times 16 \times 16$	$32 \times 16 \times 16$
4	$128 \times 8 \times 8$	$64 \times 8 \times 8$

Table 6. Dimensions of intermediate representation in the form *channels*  $\times$  *height*  $\times$  *width* used for feature-level mutual information maximization between a teacher WRN-40-2 and a student WRN-16-1 network. Alternatively, each value of  $k$  represents a pair of elements in the set  $\mathcal{R}$ .

#### A.3.2 Dissimilar CNN Architectures.

Similar approach of defining the set  $\mathcal{R}$  is followed in cases where the teacher and student networks have significantly different architectures. For instance, Table 7 shows the dimensions of intermediate representations used when the teacher network is a ResNet34 while the student is a ShuffleNetV2. Here  $k = 4$  is used, however, for some combinations of different standard architectures we use  $k = 3$  if only 3 pairs intermediate representations from the teacher and the student have the same size. Note that our method is invariant to the number of channels in the representations. Therefore, mismatch in the number of channels in pairs of representations in  $\mathcal{R}$  is inconsequential for the formulation of our losses.

### A.4. Mutual Information Discriminators

The parameterized mutual information discriminator functions ( $T_{\omega_g}$ ,  $T_{\omega_l}$ , and  $T_{\omega_f}$ ) can be modeled as neural networks. In our experiments, we use two distinct discriminator architectures inspired from the functions presented in Deep InfoMax [13].

#### A.4.1 Convolve Architecture.

In this method, the representations from the teacher and the student are concatenated together and passed through a se-

	ResNet34	ShuffleNetV2
k	$f_t^{(k)}(x)$	$f_s^{(k)}(x)$
1	$64 \times 32 \times 32$	$24 \times 32 \times 32$
2	$512 \times 16 \times 16$	$116 \times 16 \times 16$
3	$1024 \times 8 \times 8$	$232 \times 8 \times 8$
4	$2048 \times 4 \times 4$	$464 \times 4 \times 4$

Table 7. Dimensions of intermediate representation in the form  $channels \times height \times width$  used for feature-level mutual information maximization between a teacher WRN-40-2 and a student WRN-16-1 network. Alternatively, each value of  $k$  represents a pair of elements in the set  $\mathcal{R}$ .

ries of layers to get the score. For global information maximization, the final representations from both networks is concatenated together to get  $[f_s(x), f_t(x)]$ . This vector is then passed to a fully connected network with two 512-unit hidden layers, each followed by a *ReLU* non-linearity (ref. table 8). The output is then passed through another linear layer to obtain the final score.

Input	Operation	Output
$[f_t(x), f_s(x)]$	LL + ReLU	$O_1$
$O_1$	LL + ReLU	$O_2$
$O_2$	LL	score

Table 8. The architecture of the discriminator used for global information maximization. Here LL denotes Linear Layer and  $d(v)$  refers to the number of dimensions in vector  $v$ .

For local information maximization, we replicate the final representation from the teacher  $f_t(x)$  to match the  $m_K \times m_K$  size of the student’s last intermediate feature map ( $f_s^{(K)}(x)$ ). The resulting replicated tensor is then concatenated with  $f_s^{(K)}(x)$  to get  $[f_t(x), f_s^{(K)}(x)]$  which serves as the input for the critic function (ref. table 9).

Input	Operation	Output
$[f_t(x), f_s^{(K)}(x)]$	$1 \times 1$ Conv + ReLU	$O_1$
$O_1$	$1 \times 1$ Conv + ReLU	$O_2$
$O_2$	$1 \times 1$ Conv	scores

Table 9. The architecture of the discriminator used for local and feature mutual information maximization. Note that for feature mutual information maximization the input at the first layer is  $[f_t^{(k)}(x), f_s^{(k)}(x)]$ .

Similarly, consider feature mutual information maximization, for each pair in the set  $\mathcal{R}$  we use a distinct discriminator  $T_{\omega_f}^{(k)}$ . For a given  $k$ , each pair of intermediate feature representations in the set  $\mathcal{R}$  are concatenated together to get  $[f_t^{(k)}(x), f_s^{(k)}(x)]$ . Which is then passed

through two convolutional ( $1 \times 1$  kernels and 512 filters) where each layer is followed by a ReLU non-linearity. The output obtained is then further passed into a convolutional layer ( $1 \times 1$  kernels and 1 filter) to give  $m_k \times m_k$  scores (ref. table 9).

#### A.4.2 Project and Dot Architecture.

In this method, the representations from both the teacher and the student are first projected using an appropriate projection architecture with a linear shortcut. The dot-product of these projections is then computed to get the score. Positive and negative pairs of representations are passed through the discriminator to get respective scores to be passed into equation (2) to get the estimates on the lower bound of the mutual information. One-dimensional representations are projected using the architecture described in table 10, whereas for two-dimensional intermediate feature maps, projection architecture described in table 11 is used.

Input	Operation	Output
$f_t(x)$ or $f_s(x)$	LL + ReLU + LL	$O_1$
$f_t(x)$ or $f_s(x)$	LL + ReLU	$O_2$
$O_1 + O_2$	LN	$proj$

Table 10. The projection architecture used for one-dimensional inputs. Here, LL denotes linear layer while LN denotes layer normalization. Both  $f_t(x)$  and  $f_s(x)$  are projected using this architecture and their dot product is computed to get scores.

Therefore, for (1) global information maximization, both  $f_t(x)$  and  $f_s(x)$  are projected using the one-dimensional projection architecture, for (2) local information maximization, the final teacher representation,  $f_t(x)$ , is projected using the one-dimensional projection architecture and duplicated to match the size of the projected intermediate student representation projected using the two-dimensional projection architecture, a dot product of these outputs is then computed to get the scores, while for (3) feature information maximization, both representations in each pair of the set  $\mathcal{R}$  is projected using a respective two-dimensional projection architecture.

Input	Operation	Output
$f_s^{(k)}(x)$	$1 \times 1$ Conv + ReLU + LL	$O_1$
$f_s^{(k)}(x)$	$1 \times 1$ Conv + ReLU	$O_2$
$O_1 + O_2$	LN	$proj$

Table 11. The projection architecture used for two-dimensional inputs. Here, LL denotes linear layer while LN denotes layer normalization.

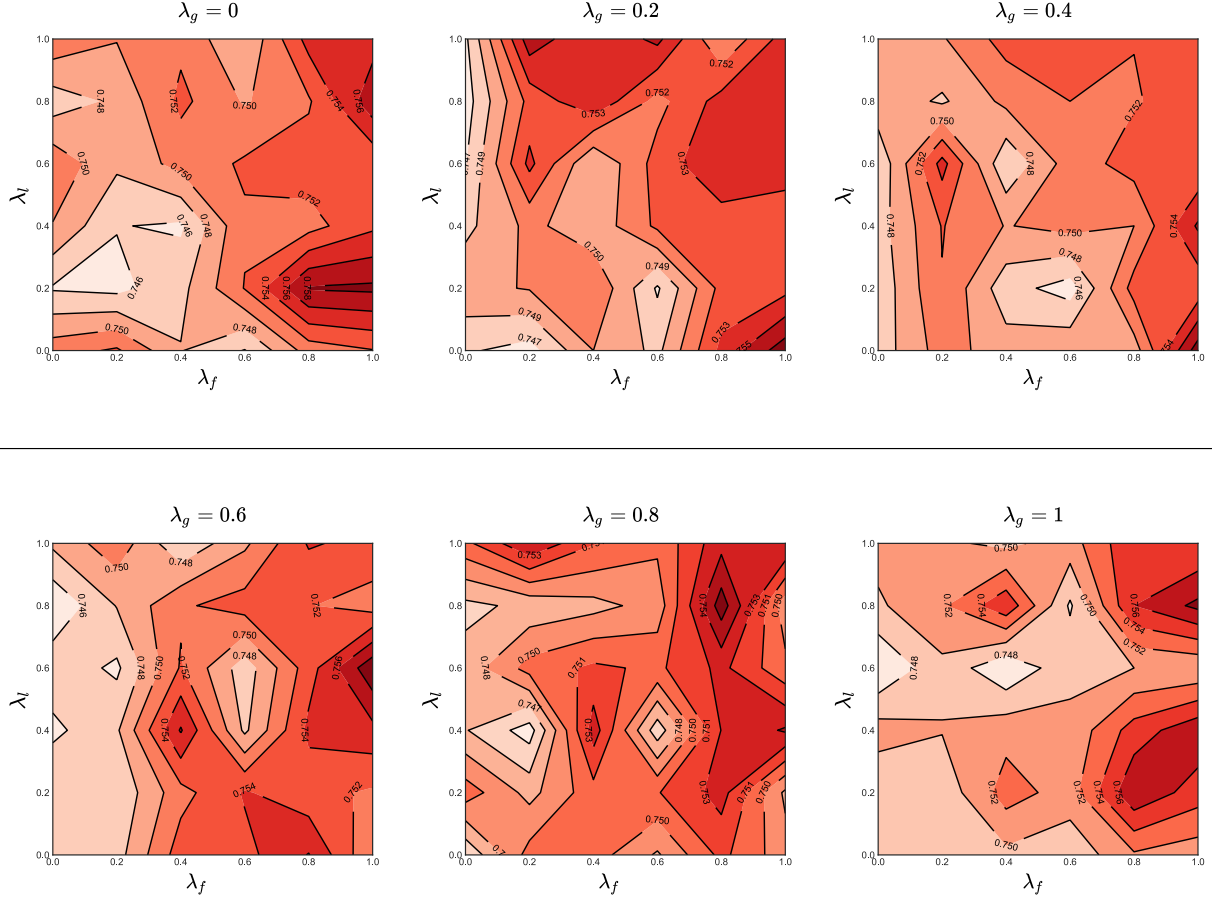


Figure 4. Results from the ablation studies on CIFAR100 dataset using a student resnet8x4 (baseline acc. 72.44%) with teacher resnet32x4 (baseline acc. 79.24%). Contour lines represent the final test accuracy of the student. Grid search was performed by varying the values of  $\lambda_f$ ,  $\lambda_g$ ,  $\lambda_l$  from 0 to 1 with increments of 0.2. In each plot, the accuracy landscape is shown with  $\lambda_g$  set to a constant value.



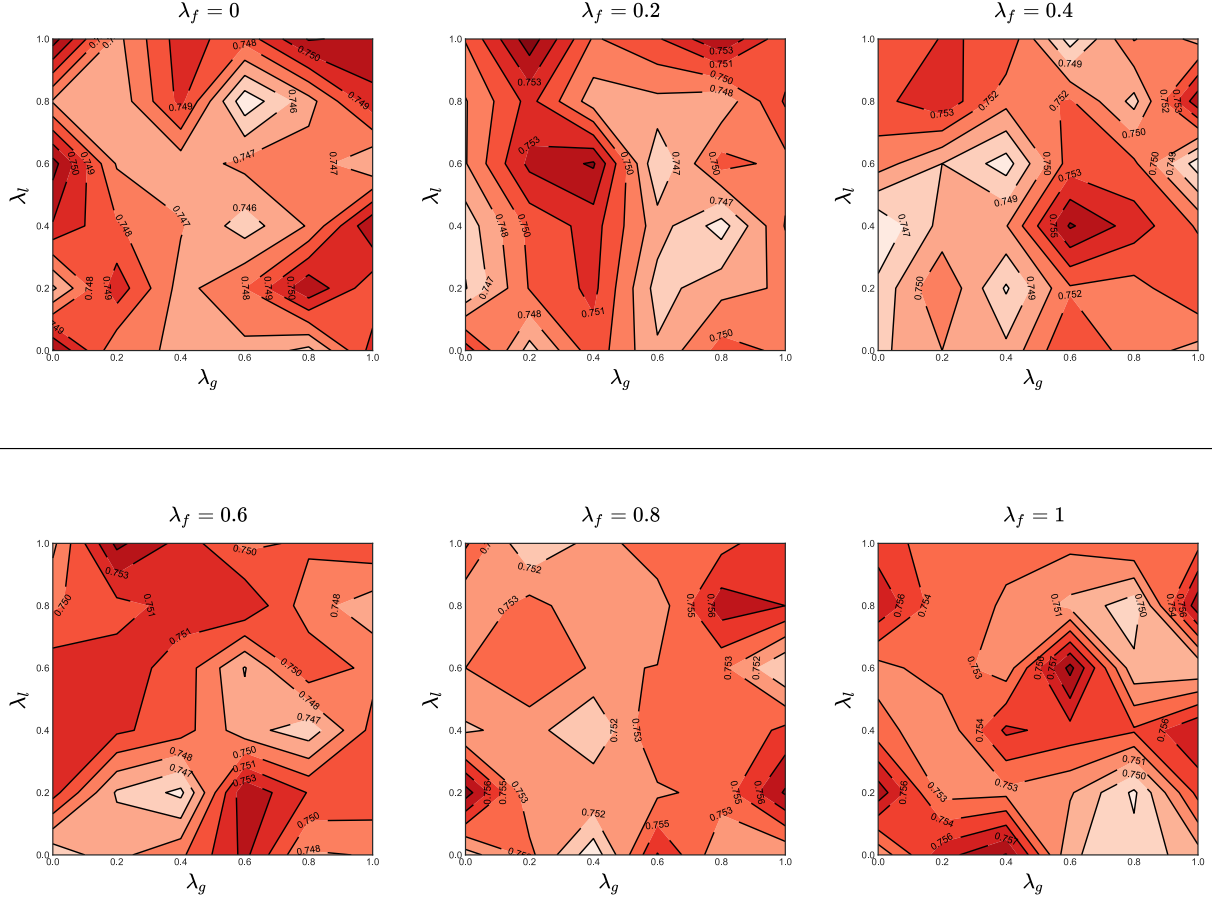


Figure 5. Results from the ablation studies on CIFAR100 dataset using a student resnet8x4 (baseline acc. 72.44%) with teacher resnet32x4 (baseline acc. 79.24%). Contour lines represent the final test accuracy of the student. Grid search was performed by varying the values of  $\lambda_f$ ,  $\lambda_g$ ,  $\lambda_l$  from 0 to 1 with increments of 0.2. In each plot, the accuracy landscape is shown with  $\lambda_f$  set to a constant value.

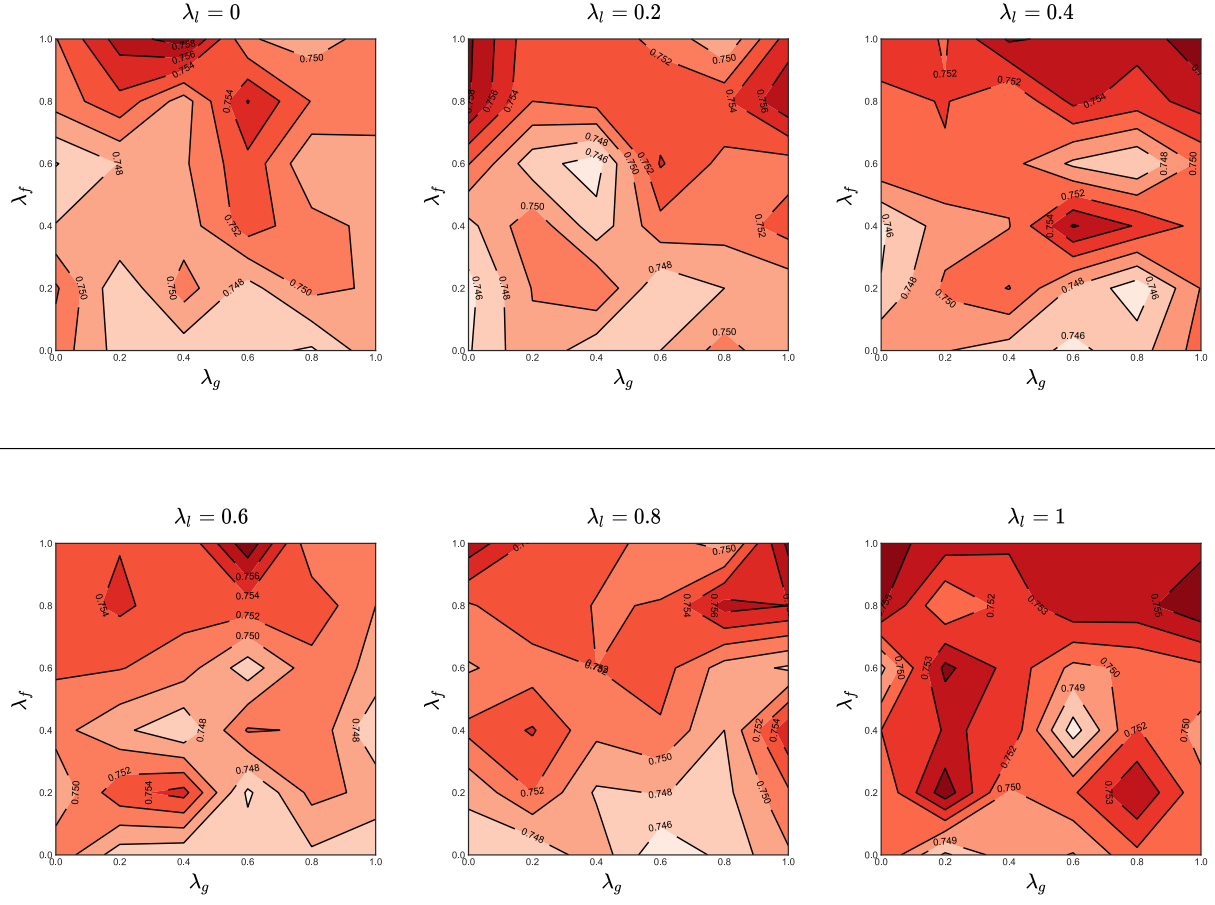


Figure 6. Results from the ablation studies on CIFAR100 dataset using a student resnet8x4 (baseline acc. 72.44%) with teacher resnet32x4 (baseline acc. 79.24%). Contour lines represent the final test accuracy of the student. Grid search was performed by varying the values of  $\lambda_f$ ,  $\lambda_g$ ,  $\lambda_l$  from 0 to 1 with increments of 0.2. In each plot, the accuracy landscape is shown with  $\lambda_l$  set to a constant value.