Conditional generative data-free knowledge distillation

Xinyi Yu\textsuperscript{1}, Ling Yan\textsuperscript{1}, Yang Yang\textsuperscript{1}, Libo Zhou\textsuperscript{1} and Linlin Ou\textsuperscript{1*}

\textsuperscript{1}Zhejiang University of Technology, Hangzhou, China.

*Corresponding author(s). E-mail(s): linlinou@zjut.edu.cn; Contributing authors: yuxy@zjut.edu.cn; lingyan@zjut.edu.cn; yangyang98@zjut.edu.cn; libozhou@zjut.edu.cn;

Abstract

Knowledge distillation has made remarkable achievements in model compression. However, most existing methods demand original training data, while real data in practice are often unavailable due to privacy, security and transmission limitation. To address this problem, we propose a conditional generative data-free knowledge distillation (CGDD) framework to train efficient portable network without any real data. In this framework, besides the knowledge extracted from teacher model, preset labels are introduced as additional auxiliary information to train the generator. The trained generator can produce meaningful samples of specified category. These samples will be employed as training data for the distillation process. In addition to using conventional distillation loss, we treat preset label as ground truth label so that the student network is directly supervised by the category of synthetic training sample. Naturally, the additional information can facilitate distillation process. Moreover, student network is forced to mimic the attention maps of teacher model and further improve its performance. We also design a novel evaluation metric called as relative accuracy to directly compare the effectiveness of different distillation methods. Trained portable network learned with the proposed data-free distillation method obtains 99.63\%, 99.07\% and 99.84\% relative accuracy on CIFAR10, CIFAR100 and Caltech101, respectively. The superiority of proposed method is demonstrated by these experimental results.

Keywords: Model compression, Knowledge distillation, CNN, Data-free
1 Introduction

In past few years, deep learning based on convolution neural networks (CNNs) has achieved state-of-the-art performance in various computer vision applications, such as image classification [1, 2], object detection [3] and semantic segmentation [4]. These techniques have been widely employed in automatic driving [5], cancer detection [6], facial beauty [7] and games intelligence [8].

Actually, the superior performance of deep learning came with over-parameterized convolution neural networks, which train on large datasets. However, ever-growing model parameters and computational cost suppress the application of high-performance models on cloud and edge devices. In this case, various effective techniques have been proposed to compress and speed-up cumbersome models, including network pruning [9–11], parameter quantization [12–14], low-rank decomposition [15, 16] and knowledge distillation [17–19]. To be specific, network pruning can compress a heavy model by removing redundant weights and filters that have less information, while parameter quantization compresses a cumbersome network by reducing the number of bits required to represent network parameters. Moreover, convolution computation is the most complicated operation in deep models, so low-rank decomposition estimates and decomposes the filters by using matrix or tensor decomposition techniques.

Knowledge distillation (KD) is a knowledge transfer framework to train a small “student” network under the supervision of a large pre-trained “teacher” network. It can compress model regardless of the network structure difference. Hinton et al. [17] transferred the knowledge from teacher model to student network by minimizing the discrepancy between the outputs. To extract more information about the relationships between different categories, they proposed “SoftMax temperature” to soften model output. In order to improve distillation performance, further work utilizes the latent information hidden in intermediate layers. FitNets [18] trains student network by utilizing both the outputs (logits) and intermediate features from teacher network. Zagoruyko et al. [19] promoted student performance by forcing student to mimic the attention maps of teacher model.

KD has extensive applications in Computer Vision (CV), Natural Language Processing (NLP), speech recognition, etc. It is very efficient for compressing cumbersome models when original or alternative data is available. However, training datasets for given model in real-world applications are often unavailable due to privacy, security and transmission limitations. For instance, most users do not want their electronic records containing private information to be leaked to others, or some training datasets are too huge to transmit in time. Due to the absence of original training data, conventional model compression methods cannot be directly applied for learning compact and efficient deep models. For this issue, an effective way is data-free knowledge distillation [20–23], which does not require any real data. Recently, several data-free (zero-shot) knowledge distillation methods that exploit generative adversarial networks (GAN) have been proposed for compressing heavy models without any real
Fig. 1 The overall framework of proposed conditional generative data-free knowledge distillation method. In this framework, we train the generator to produce meaningful samples and use the samples as training data. Subsequently, we combine the student and teacher network together as discriminator, and train student network under supervision of teacher model. The alternating training process of student network and generator is adversarial.

training data. In practice, due to the lack of real data, conventional generation techniques such as GAN [24] and VAE [25] cannot be directly applied on data-free KD. To address this problem, Chen et al. [21] designed a novel framework to train a generator to create one-hot images, which can highly activate the neurons in teacher model. Fang et al. [22] presented an adversarial distillation mechanism to move generated samples towards the areas where current student has not been well trained. Luo et al. [23] leveraged the intrinsic batch normalization statistics of pre-trained teacher model to train generator and achieved state-of-the-art results.

Although existing work has basically realized data-free knowledge distillation, there is still a performance gap between the trained student and teacher networks, while the student model trained by conventional data-driven distillation methods achieves even better performance than teacher model. Moreover, the trained generator can only produce images randomly, but cannot control the category of synthetic images. In this situation, synthetic images are class-balanced and may be incompatible with long-tailed distribution tasks. For these problems, we propose a conditional generative data-free distillation framework to attain efficient data generation and model compression, as illustrated in Fig. 1. We aim to train a strong generator by exploiting the latent knowledge from pre-trained teacher network, and then treat the generated samples as training data to train student model. To be specific, in addition to using the knowledge extracted from teacher model, we also introduce preset labels as additional auxiliary information to train the generator and make the trained generator produce images of specified class rather than random category. Since the category of generated images is determined by preset label, student network can be trained under the supervision of ground truth labels
(preset labels) when the synthetic image is employed as training data. In data-driven distillation methods, ground truth labels are also used as constraint to train student network. With the supervision of preset labels, we narrow the discrepancy between data-free and data-driven distillation methods. In order to further improve distillation performance, we force student network to mimic the attention maps of teacher model and improve its performance.

To demonstrate the effectiveness of proposed method we conduct extensive experiments on several datasets including MNIST, CIFAR10, CIFAR100 and Caltech101. Then, a series of ablation experiments are performed to analyze different components in our method. The main contributions of this paper can be summarized as follows:

- A conditional generative data-free distillation framework is proposed, which can consistently produce meaningful training samples for efficient knowledge distillation.
- We train a generator that can control the category of synthetic images by introducing auxiliary class constraint and convert the training process of generator into a semi-supervised process.
- We achieve attention transfer in data-free distillation task, and greatly improve the performance of student model.
- We demonstrate the effectiveness of proposed data-free distillation method on several datasets and achieve state-of-the-art results on MNIST, CIFAR100 and Caltech101.

2 Related work

In this section, we briefly review existing works on related topics.

2.1 Data-driven knowledge distillation

Knowledge distillation (KD) is a general technique that can be applied for model compression. It refers to the method that helps the training of a smaller student network under the guidance of a teacher network. KD can compress a network regardless of the structural difference between student and teacher networks. In KD, the knowledge can be transferred by minimizing the difference between logits (the inputs of last SoftMax layer) produced by student and teacher networks. In many situations, however, the output of teacher has very high probability at correct class, while the probabilities at other classes are very close to zero. In this case, the logits cannot provide more information than one-hot label. To tackle this problem, Hinton et al. [17] introduced the concept of “SoftMax temperature” to soften the output of network and extract more information about the relationships between different categories. Moreover, Furlanello et al. [26] trained a set of student networks with logits transfer and integrated them into the final student model. Mirzadeh et al. [27] introduced an intermediate-sized network (teacher assistant) as the bridge and trained the student network based on the teacher assistant.
Besides the logits from teacher network, transferring knowledge to student network can be achieved by utilizing the feature information in hidden layers. Romero et al. [18] proposed FitNets to train student network using both the outputs (logits) and the intermediate features learned by teacher network to improve student performance. Zagoruyko et al. [19] transformed the feature in hidden layers into attention map and improved student performance by forcing student network to mimic the attention map of teacher network. Yim et al. [28] introduced FSP (Flow of Solution Procedure) matrix and forced student model to mimic the relationship between input and output of teacher network. Park et al. [29] proposed a relational knowledge distillation method to transfer the mutual relations of data based on distance-wise and angle-wise loss. Bai et al. [30] designed a novel layer-wise cross distillation method to achieve few-shot model compression. Li et al. [31] added the supervision from proposal features and trained a portable objective detection model. Liu et al. [32] proposed a structure distillation method for dense prediction. They extracted the holistic knowledge and pairwise similarity of the network by using adversarial learning and building static graph, respectively. Although KD can downsize a network regardless of the structural difference, several existing works have explored the effect of structure on distillation performance [33, 34]. Furthermore, many model compression methods were suggested by combining knowledge distillation with other techniques, such as NAS [34], RL [35] and GNN [36]. [37] made a review and outlook for the development of knowledge distillation. Knowledge distillation has achieved good results, while there are still challenges in many fields, such as cross-domain task, cross-modal task and data-free distillation. It remains very important to explore more efficient knowledge representation and utilization methods.

2.2 Data-free knowledge distillation

Real training data is often unavailable in real world due to the data privacy and security issues. Data-free model compression has been a hot topic and draw more and more attention in recent years. To tackle this problem, many data-free compression techniques have been proposed, including data-free network pruning [38, 39], parameter quantization [40–42] and knowledge distillation [20–23, 43].

In data-free knowledge distillation, Lopes et al. [20] first attempted to reconstruct original data from metadata and utilized the synthetic data to train student network. However, this method still requires metadata which extract from real data. Nayak et al. [43] modeled the SoftMax space as a Dirichlet distribution and updated random noise images to craft Data Impressions without metadata. Chen et al. [21] designed a novel framework (DAFL) to train a generator which match the distribution of original data. The teacher model is fixed as discriminator. However, vanilla loss in GAN is invalid because the output of trained teacher is category rather than image score. To this end, DAFL designed several novel loss functions to guide the optimization process of generator. In contrast, Fang et al. [22] presented an adversarial distillation
mechanism (DFAD) to craft a portable student network. In DFAD, student and teacher models jointly play the role of discriminator. Student network is trained to reduce the output discrepancy, while generator is trained to create “hard samples” and enlarge the discrepancy. Yin et al. [44] synthesized images by model inversion without additional generator, and improved the quality of images by utilizing batch normalization statistics. To achieve data-free distillation on large-scale datasets, Luo et al. [23] trained one generator for each class. It gets remarkable performance gain and avoids mode collapse (generator produces similar images and thus loses image diversity) in image generation, but requires huge computational resources. Meanwhile, Jin et al. [45] introduced a simple fully connection network as discriminator to distinguish teacher and student models. Fang et al. [46] improved image diversity by introducing contrastive learning.

Although the methods mentioned above have basically realized data-free distillation, they did not make full use of the latent knowledge from teacher network. In this case, student model is difficult to obtain comparable performance to data-driven distillation methods. In addition, the trained generator cannot produce images of the specified category as required, which is incompatible with long-tailed distribution data. Therefore, a more efficient approach for data-free image generation and distillation is required.

2.3 Generative Adversarial Networks (GAN)

In order to transfer knowledge from teacher to student network without real training samples, we adapt the generative adversarial networks to attain efficient data generation. Generative adversarial network [24] is a deep learning framework which contains two interacting neural networks: a generator $G$ that captures the distribution of real data, and a discriminator $D$ to distinguish the real and fake samples (generated by $G$). Two networks are trained jointly through a minimax game. To be specific, given a set of random noise vectors $z$ sampled from distribution $p_z$, $G$ maps $z$ to the distribution of desired data $x$, i.e. $G : z \rightarrow x$. $D$ outputs a high score (close to 1) for real sample and produces low probability (close to 0) for generated sample. With adversarial training, $G$ can generate meaningful data that match real data distribution. Moreover, in order to make generator produces images of specified class, Mirza et al. [47] proposed the conditional GAN, which applies class condition to both generator and discriminator networks. To avoid training instability and mode collapse in conventional GAN, Arjovsky et al. [48] proposed Wasserstein GAN (WGAN) to make training more stable, and Qi et al. [49] regularize the adversarial loss in terms of the Lipschitz regularization.

GAN has demonstrated powerful capabilities in image translation [50], super-resolution [51], restoration [52], etc. The powerful capabilities theoretically could be applied for data-free sample generation and then realize data-free knowledge distillation. However, all these methods require real data to train the generator so as to produce realistic images. In this case, it is significant to explore the implementation of GAN when real data is unavailable.
In conclusion, we aim to a strong generator by utilizing the knowledge from teacher network. The images generated by generator will be treated as training samples for data-free knowledge distillation.

3 Method

In this section, we will describe the details of proposed conditional generative data-free knowledge distillation framework. As shown in Figure 1, the framework contains three networks: a pre-trained teacher network $T$, a smaller student network $S$, and a generator $G$. A set of random noise $Z = \{z^1, z^2, ..., z^n\}$ is inputted into generator $G$ to craft training samples $G(z)$, where $n$ is the number of samples, $z^i \in R^m$ is sampled from normal distribution $p_z$.

3.1 Data-free conditional generator

As mentioned above, real data of pre-trained teacher model is often unavailable for various concerns. To tackle this problem, training student network by utilizing alternative data [53] or irrelevant data is reasonable. However, collecting alternative data is still onerous, while irrelevant data will drastically deteriorate the performance of student network. In this case, an effective way is to reconstruct data as training samples. Although pre-trained teacher model contains rich knowledge of original training data, it is difficult to exploit the latent knowledge and to reconstruct meaningful samples. Fortunately, generative adversarial network offers a powerful capability to exploit data distribution and craft images. We produce training samples and realize data-free distillation based on the mechanism of GAN.

In existing data-free distillation methods, the trained generators can only randomly produce images and cannot control the category of generated images. The generator has an equal probability to generate each class of images, so the synthetic training images are class-balanced. Class-balanced synthetic data is desirable for general teacher model which is trained on class-balanced datasets. But for the teacher which is trained on long-tailed datasets, it may cause the failure of the special design for long-tailed data, and result in poor performance of distillation. To this end, we introduce preset class labels as auxiliary information to build a conditional generator. Under the supervision of class condition, trained generator can produce images of specified category on request and meet different requirements. To be specific, we uniformly sample a set of preset labels $\{y^i\}$ from $\{0, 1, ..., c - 1\}$, where $c$ is the number of classes. The teacher network is expected to produce the same prediction result as the preset label, when preset label is fed into generator to craft training sample. In this case, classes matching loss can be formulated as:

$$L_{CM} = \frac{1}{n} \sum_{i=1}^{n} H(\text{softmax} (l^i_T), y^i)$$ \hspace{1cm} (1)
where \( y^i \) is preset label of \( i \)-th image, \( H(\cdot) \) denotes cross-entropy loss, \( l_T = T(G(z \mid y)) \) is the logits of teacher network. This loss function calculates cross-entropy between the SoftMax outputs of teacher and preset labels. It is minimal when predicted results are the same as preset labels. When teacher network is fixed, we force the generator to match preset class by reducing the loss function. With classes matching loss, the training process of generator is transformed from unsupervised to a semi-supervised process.

In practice, we do not optimize this function alone. To measure the class balance of generated images, we introduce information entropy loss and this function is complementary to the classes matching loss function in Eq. 1. The information entropy loss is defined as:

\[
L_{ie} = -H_{info}(\bar{p}) = \frac{1}{c} \sum_{i=1}^{c} \bar{p}_i \log(\bar{p}_i) \quad (2)
\]

where \( \bar{p} = \frac{1}{n} \sum_j l^j_T \) is the average logit of all samples, \( H_{info}(\cdot) \) is information entropy. When this function takes the minimum, each element in \( \bar{p} \) equals to \( \frac{1}{c} \). In this case, we can force generator to produce a balanced set of synthetic samples by minimizing this loss function. Moreover, for a classification task where training image dataset is class-balanced, Eq. 2 can be utilized together with Eq. 1 to jointly promote the class-balance of generated samples. But for the training dataset which is class-imbalanced, information entropy loss should be discard, and only classes matching loss will be employed to constrain the category of synthetic images.

Besides information entropy, teacher output is also important information of input images. In classification task, high performance teacher model produces one-hot like vectors for input images. If synthetic samples follow the same distribution as original training data, teacher model will also output one-hot like vectors. In other words, the input image will match original data distribution when the teacher outputs a one-hot vector. To this end, we compute the cross-entropy between predicted label and logits of teacher network, and formulate the one-hot loss as:

\[
L_{oh} = \frac{1}{n} \sum_{i=1}^{n} H(l^i_T, y^i_T) \quad (3)
\]

where \( y^i_T = \arg\max_j (l^i_T)_j \) is predicted label of teacher model. With supervision of one-hot loss, generator is forced to craft images that matching original training dataset. This loss function can be used together with classes matching loss to help the training of generator. Eq. 1 requires the synthetic image to be a specified category, while Eq. 3 forces the image has more prominent feature of a certain category. Since information entropy loss and one-hot loss have no clear optimization objective, they can be combined as unsupervised loss:

\[
L_{US} = L_{oh} + \lambda_{ie} L_{ie} \quad (4)
\]
where $\lambda_{ie}$ is scale weight for balancing different components.

Loss functions in Eq. 1 and 4 only utilize the knowledge from teacher model. In data-free distillation, however, student network will train together with generator. Naturally, the information from student model is beneficial for training generators that are compatible with student model. To this end, we introduce the discrepancy estimation loss of student and teacher predicted results for synthetic images. In DFAD, Fang et al. [22] stated that KLD and Mean Square Error (MSE) produce gradient decay when student network converges on synthetic images. Then gradient decay deactivates the learning of generator and makes the minmax game over. In this work, mean absolute error (MAE) of two network outputs is used as discrepancy estimation loss:

$$L_{DE} = \mathbb{E}_{z \sim p_z(z)} \left[ \frac{1}{n} \| T(G(z \mid y)) - S(G(z \mid y)) \|_1 \right]$$

(5)

where $T(G(z \mid y))$ and $S(G(z \mid y))$ are the output of teacher and student networks for synthetic image $G(z \mid y)$. In imitation stage, student model should produce the same predictions as teacher for synthetic images and minimize the discrepancy estimation loss. However, if this function is applied as optimization objective, generator tends to produce “easy samples” which are easy to classify. With these “easy samples”, student network can easily learn the shallow knowledge of teacher and produces same predictions. But the deep latent knowledge cannot be transferred to student network without exploring “hard samples” (both teacher and student networks are difficult to distinguish). To tackle this problem, we conversely maximize this loss to move generator towards the areas where current student network has not been well trained.

By combing the aforementioned loss functions in Eq. 1, 4 and 5, the generation loss can be defined as:

$$L_G = -L_{DE} + \lambda_{US}L_{US} + \lambda_{CM}L_{CM} + \lambda_{BNS}L_{BNS}$$

(6)

where $L_{BNS}$ [23] is the loss of batch-norm statistics. $\lambda_{US}$, $\lambda_{CM}$ and $\lambda_{BNS}$ are scale weights that trade-off different loss terms. In generation stage, generation loss is employed as the optimization objective of generator. Then the trained generator can craft training samples that match real data and student network.

### 3.2 Distillation process based on synthetic images

In section 3.1, discrepancy estimation loss is employed to measure the discrepancy between the outputs of teacher and student networks. During generation stage, this loss is maximized so that generator produces “hard samples”. In distillation process, we minimize this loss function conversely to train the student model. Thus an adversarial learning framework is constructed. To minimize the discrepancy estimation loss, student network will learn deep latent knowledge
and produces the same output as teacher network. In the adversarial training framework, “hard samples” will be consistently produced so that student network explores unseen regions and achieves better performance.

Besides estimating discrepancy, ground truth label can be employed as constraint to optimize student network if it exists. It is a general method applied in data-driven distillation. However, ground truth label in data-free task is unavailable, causing the constraint to fail. To facilitate distillation process and narrow the gap between data-free and data-driven distillation methods, we treat the preset label in Eq. 1 as ground truth label and define ground truth loss as:

$$L_{GT} = \frac{1}{n} \sum_{i=1}^{n} H(\text{softmax}(l^i_T), y^i)$$  \hspace{1cm} (7)

where $l^S = S(G(z \mid y))$ is logits of student network. Different from Eq. 1, here we calculate the cross-entropy loss between student predictions and preset labels. By introducing this loss, student network can be optimized under the supervision from the ground truth labels of training data. Although discrepancy estimation loss and ground truth loss provide useful information, it is insufficient for student network to complete complex task. In addition, student model cannot learn useful knowledge from teacher, when the capacity gap between teacher and student networks is too huge. Data-free distillation of a strong teacher is even more difficult due to the absence of real data. To this end, attention transfer loss is introduced to extract the knowledge hidden in intermediate layers. In natural world, attention plays a critical role in human visual experience, and it makes a system pay more attention to an object. [19] claimed that the attention map in artificial neural network shows the area that it focuses on. Larger attention value denotes the higher probability that the pixel belongs to the target. Attention map based on activation is formulated as:

$$F(A) = \sum_{i=1}^{C} |A_i|^2$$  \hspace{1cm} (8)

where $A$ denotes activation of each channel, $C$ is the number of channels. Power and absolute value operations are element-wise. Then the attention transfer loss can be formulated as:

$$L_{AT} = \sum_{j \in I} \left\| \frac{Q^j_S}{\|Q^j_S\|_2} - \frac{Q^j_T}{\|Q^j_T\|_2} \right\|_2$$ \hspace{1cm} (9)

where $Q^j_S = \text{vec} \left( F(A^j_S) \right)$ and $Q^j_T = \text{vec} \left( F(A^j_T) \right)$ are the $j$-th pair attention maps of student and teacher networks. $\text{vec}(\cdot)$ converts feature maps into vector form, $I$ is the number of layers that selected to perform attention transfer. By minimizing this function, we can force student network to mimic the attention maps of teacher network and obtain better performance. After
combining the loss functions in Eq. 5, 7 and 9, we obtain the distillation loss:

\[ L_{KD} = L_{DE} + \lambda_{GT} L_{GT} + \lambda_{AT} L_{AT} \]  \hspace{1cm} (10)

where \( \lambda_{GT} \) and \( \lambda_{AT} \) are trade-off hyper-parameters.

In distillation stage, we use the generator \( G \) to craft a set of images \( G(z \mid y) \) as training data. Then the teacher network \( T \) is fixed and only the parameters in student network \( S \) are updated. In previous works, data-free distillation method has a big gap with data-driven distillation method due to the insufficient of knowledge and the absence of truth label. With ground truth loss and attention transfer loss, we overcome the difficulties and attain efficient data-free distillation.

### 3.3 Conditional generative data-free knowledge distillation

The complete conditional generative data-free knowledge distillation framework for learning efficient portable network is summarized in Algorithm 1. Our algorithm includes three networks: a pre-trained teacher network \( T \), a student network \( S \), and a generator \( G \). The training procedure can be divided into two stages, namely generation stage and distillation stage. During training, teacher network \( T \) is fixed, we alternately update the parameters in student network \( S \).
and generator $G$ based on gradient descent. In image generation stage, we fix student network $S$ and update the weights of generator $G$ by minimizing Eq. 6. The trained generator $G$ produces meaningful hard samples $G(z \mid y)$ with the same distribution as original training data. In distillation stage, generated samples are employed as training data to adjust the parameters in student model $S$. Eq. 10 is minimized by forcing student network to mimic teacher network $S$ so as to achieve better student performance. Different from training the generator, the student network is updated $k$ times in each epoch to ensure its convergence.

Throughout the adversarial distillation process, generator $G$ continuously produces meaningful samples, and student network $S$ consistently learns useful knowledge from teacher network. The two processes are repeated alternately until both networks converge. At this time, we will obtain an efficient portable student model.

4 Experiments

In this section, we conduct extensive experiments on different datasets to validate the effectiveness of proposed data-free distillation method. Furthermore, a series of ablation experiments are performed to analyze different components in proposed method.

4.1 Datasets and Models

We validate the proposed distillation method on following datasets: MNIST, CIFAR10, CIFAR100, and Caltech101.

**MNIST.** MNIST [54] is a well-known image dataset of handwritten digits, which is composed of $28 \times 28$ pixels gray level images. It is very simple and only contains 10 classes (from 0 to 9) with 60,000 training images and 10,000 testing images. These images have been normalized and digits located in the center of images. We adopt LeNet-5 [55] as teacher network and a LeNet-5-Half [21] as student network.

**CIFAR10 and CIFAR100.** CIFAR10 and CIFAR100 [56] are slightly more complex image datasets. Both of them contain 50,000 training samples and 10,000 testing samples. CIFAR10 consists of $32 \times 32$ pixels color images with 10 categories. CIFAR100 has the same image size and contains 100 categories. In the experiments of CIFAR, ResNet-34 [57] and ResNet-18 are selected as teacher and student networks, respectively.

**Caltech101.** Caltech101 dataset [58] consists of 101 categories. Each category contains 40 to 800 images, and most categories have about 50 images. In training process, we randomly split 80% images into training data, and 20% images into testing data. Also, the images are resized and center cropped to $128 \times 128$ pixels during training. To generate larger images, we adopt a generator which is stronger than that use on CIAFR and MNIST. As with the experiments on CIFAR datasets, ResNet-34 and ResNet-18 networks are employed on the experiments of Caltech101 dataset.
4.2 Experimental details

We implement all networks and distillation methods with PyTorch, and do not use any data augmentation technique for making fair comparison. For all datasets, SGD with momentum 0.9 and weight decay $5 \times 10^{-4}$ is applied to update the student network, and Adam is employed to optimize the generator. Moreover, we utilize the generators following [22] and make some adjustment to fit proposed method.

4.2.1 Talent selection strategy

In practice, if it is impossible to determine whether the methods or hyper-parameters are effective, performing the complete training for models will lead to resources waste. For reducing computational cost and training time, we propose a simple training strategy, that is talent selection strategy. It is a manual tuning technique that helps us verify the effectiveness of new method and adjust hyper-parameters with minimal time consumption. To be specific, if the model requires 200 training epochs and twice learning rate adjustment, we train it 50 epochs and change learning rate once at 40th epoch. Then we conduct a series of experiments with different components or hyper-parameters. In this case, the effectiveness of current components or hyper-parameters can be verified by comparing the results of different settings. In talent selection process, the influence of each component or hyper-parameters will be clearly observed, and we can tune the hyper-parameters according to human prior knowledge. Also, whether the current adjustment is effective can be estimated based on the variation trend of training loss and testing accuracy. If the performance with current hyper-parameters is much worse than that in other settings, we can interrupt current training process in advance so as to further reducing resource waste. When the performance changes suddenly, we will decide to interrupt or continue the training process according to our experience. All the actions mentioned above can greatly reduce training time and computational cost.

After accomplishing talent selection training with different hyper-parameters settings, 2-4 groups better settings will be chosen for complete training. Then the settings with best result in complete training will be selected as final hyper-parameters. Although the final hyper parameters may not be globally optimal for the task, it is a balance of network performance and computational cost. It should be noted that even if the best settings obtained in talent selection may not win the final, they are able to make models converge faster. In fact, if target task does not require a very high precision, we can employ these settings to obtain suboptimal but faster convergence models.

4.2.2 Evaluation

In classification tasks, prediction accuracy is often applied to evaluate different methods, while the accuracy gap between teacher and student models is usually employed to appraise the effectiveness of different distillation techniques.
Table 1 Prediction accuracy and relative accuracy of different methods on MNIST dataset. Teacher models have different accuracy in different works.

| Method   | Teacher Acc.(%) | Student Acc.(%) | Relative Acc.(%) |
|----------|-----------------|-----------------|-----------------|
| DAFL [21]| 98.91           | 98.20           | 99.28           |
| DFAD [22]| 98.97           | 98.30           | 99.32           |
| DFKD [45]| 98.97           | 98.45           | 99.53           |
| Ours     | 98.97           | 98.62           | 99.65           |

However, teacher models in different distillation work are diverse and have different accuracy. In this situation, it is difficult to directly compare the effectiveness of different distillation methods. To this end, we propose an additional evaluation metric and define it as relative accuracy:

\[
A_{rel} = \frac{A_S}{A_T} \times 100\%
\]  

where \(A_S\) and \(A_T\) are the accuracy of student and teacher networks. With relative accuracy, we can directly compare different distillation methods, thus avoiding the impact of using different teacher models.

4.3 Experimental results

4.3.1 Experiments on MNIST

We first implement proposed method on MNIST which is composed of 28×28 pixels. In training process, the images is resized to 32×32 pixels and batch size is 512. The initial learning rates are 0.01 and 0.001 for student and generator, respectively. For the experiments on MNIST, training epoch is set as 60 and learning rates are decayed once at 50th epoch. In each epoch, we iterate the training process 50 steps and set \(k\) as 5. In addition, LeNet-5 architecture does not contain batch-norm layers, so the batch-norm loss \(L_{BNS}\) should be discard.

In this experiment, student network will quickly converge to a high accuracy, and the generator can produce meaningful images after several training. For simple task such as MNIST, we should pay more attention to training epoch and avoid the models over-fitting. It is worth noting that, data-free generator tends to produce semantic images due to the lack of supervision of real images, while conventional GAN will produce natural images.

Table 1 reports the results of different data-free distillation method on MINIST dataset. With the absence of real training data, DAFL [21], DFAD [22] and DFKD [45] obtained 99.28%, 99.32% and 99.53% relative accuracy, respectively. After completing adversarial training, the student network in our method achieves 98.62% prediction accuracy and 99.65% relative accuracy while teacher network has 98.97% prediction accuracy. Our method achieves a little but non-negligible gain when accuracy is close to upper bound. Excellent results show the effectiveness of the proposed method.
Table 2 Experimental results of different methods on CIFAR. We achieve comparable or even better performance on CIFAR10 and state-of-the-art results on CIFAR100. It shows the superiority of our method.

| Method   | CIFAR10 | CIFAR100 |
|----------|---------|----------|
|          | Teacher | Student  | Relative | Teacher | Student  | Relative |
|          | Acc. (%)| Acc. (%) | Acc. (%) | Acc. (%)| Acc. (%) | Acc. (%) |
| DAFL     | 95.58   | 92.22    | 96.48    | 77.84   | 74.47    | 95.67    |
| DFAD     | 95.54   | 93.30    | 97.65    | 77.50   | 67.70    | 87.35    |
| LS-GDFD [23] | 95.05 | **95.02** | **99.97** | 77.26 | 76.42 | 98.91 |
| CMI [46] | 95.70   | 94.84    | 99.10    | 78.05   | 77.04    | 98.97    |
| DDAD [59] | 95.54   | 94.81    | 99.23    | 77.50   | 75.04    | 96.83    |
| DFKD     | 95.32   | 92.84    | 97.40    | 77.28   | 74.45    | 96.34    |
| Ours     | 95.54   | **95.19** | **99.63** | 77.52   | **76.80** | **99.07** |

4.3.2 Experiments on CIFAR

CIFAR10 and CIFAR100 datasets are composed of 32×32 pixel images and have 10 and 100 categories, respectively. For CIFAR datasets, the initial learning rate of student network and generator are 0.1 and 0.001, respectively. In training process, the training epoch is 250, batch size is 256. Also, we train student network and generator 100 steps, and decay learning rates twice. Under this setting, student network can see 256×100×5 (K=5) synthetic images in each epoch. It is even more than the images in original CIFAR datasets. In adversarial distillation framework, student model can sufficiently explore the data space and be fully trained. Since CIFAR dataset is more complex than MNIST, complete training process will take a lot of time. To this end, we adopt talent selection strategy to tune hyper-parameters and train the models. To be specific, we employ 70 epochs and 50 steps for talent selection training while complete training requires 250 epochs and 100 steps. The learning rates will be decayed once at 50th epoch. It should be noted that, although training epochs and learning rates in CIFAR datasets are identical, hyper-parameters in two datasets need be adjusted according to the experimental results.

After talent selection process, three better settings of hyper-parameters will be chosen for complete training. Table 2 summarizes experimental results of different work on CIFAR datasets. On CIFAR10, LS-GDFD [23] proposed by Google Research achieves best results of 99.97% relative accuracy and 95.02% student accuracy. Their results are only 0.03% less than teacher accuracy, which is amazing. According to their paper, LS-GDFD perform distillation for 60K epochs with batch size of 32768 on CIFAR datasets, while we set batch size to 256 because of the limitations of computational resources. Even so, the trained student network obtains 95.19% student accuracy and 99.63% relative accuracy on CIFAR10. For CIFAR100, we achieve 76.80% student accuracy and 99.07% relative accuracy which even outperforms the experimental results of LS-GDFD. This suggests that our method is not only applicable for simple task, but also for complex datasets and networks. It is worth noting that, LS-GDFD trains one generator for each category of images to avoid mode.
Fig. 2 Synthetic images on CIFAR10 (top) and CIFAR100 (bottom), and each row has same category. Left: Real images sampled from real datasets. Right: Images generated by the conditional generator. Notably, since no real data is provided, the generator tends to produce semantic images rather than natural images.

Table 3 Student accuracy and relative accuracy on Caltech101. We obtain the state-of-the-art results.

| Method | Teacher Acc.(%) | Student Acc.(%) | Relative Acc.(%) |
|--------|-----------------|-----------------|------------------|
| DFAD   | 76.60           | 73.50           | 95.95            |
| DDAD   | 76.60           | 75.01           | 97.92            |
| Ours   | 76.58           | **76.46**       | **99.84**        |

collapse and improve student accuracy. Their method is useful but training a large number of generators requires heavy computational resources. For fair comparison, we only compare with the results obtained with single generator. Fig. 2 shows the synthetic images on CIFAR datasets.

4.3.3 Experiments on Caltech101

In order to further validate the effectiveness of proposed method, we extend experiments to other high-resolution datasets. Caltech101 contains 101 categories and each image is larger than 128×128. In training process, the images are resized to 128×128. Batch size and training epoch are 128 and 300, the initial learning rates of student model and the generator are 0.05 and 0.001, respectively. Similar to the experiments on CIFAR datasets, talent selection strategy is applied on Caltech101. In talent selection training process, we train models 70 epochs to search optimal hyper-parameters and explore the effects of each component. For generating larger images, we adjust the generator in DCGAN [60] to fit our method so as to produce meaningful training images.

Table 3 reports the experimental results on Caltech101. With the same teacher accuracy, DFAD and DDAD obtained 73.5% and 75.01% prediction accuracy, respectively. Compare with them, our method achieves best student
accuracy of 76.46% and highest relative accuracy of 99.84%. It demonstrates that our method is not only adapted to low-resolution images but also can extended to other tasks with larger images. Fig. 3 shows the gradient-weighted class activation heatmaps of teacher and student models, where the teacher model is trained on real training data and the student network is trained with synthetic images. The visualization results show that student model has well learned the knowledge from teacher network so as to extract local and global features of the images well. Also, it is proved that the feature-based distillation method is effective for data-free distillation.

4.4 Ablation experiments

In above sections, we have conducted extensive experiments on different datasets. The experimental results demonstrate our method can be applied to different data-free distillation tasks. However, our method contains many different components, and each component has different role. Besides the loss functions that proposed in previous works, three novel loss functions are introduced in our conditional generative data-free distillation framework, that is, ground truth loss (GT), classes matching loss (CM) and attention transfer loss (AT). In the proposed distillation method, GT loss makes the student network be supervised by preset labels, and it works only if the category of synthetic images is controlled by CM loss. In other words, GT loss is not independent of CM loss. To further explore the role of each function, we perform a series of ablation experiments on CIFAR100 and follow the experimental settings on section 4.3.2. Table 4 summarizes the results of various design components.

First, we introduce attention transfer to the distillation framework. Student accuracy without any novel loss function is 74.81%, and it is improved to 76.34% with the introduction of AT loss. In other words, we get 1.53% performance gain, which suggests that the attention transfer can greatly improve distillation performance. Actually, a pre-trained teacher model will focus on target area of input images and capture useful information. With attention
transfer, student network can learn the knowledge from teacher and achieve better performance. Besides the attention transfer, there also exist many efficient feature-based distillation method. Since ablation study has demonstrated the effectiveness of feature-based distillation method, we will improve data-free distillation task by exploring other feature-based distillation techniques. Though existing method is useful, the challenges remain, such as the representation of knowledge and the selection of distillation position. In knowledge distillation, feature map includes rich and varied information. Generally, the knowledge should be converted into a simple form so that the student network can learn easier. Also, the transformation process should minimize the loss of information. To this end, a more effective knowledge representation method is required. Furthermore, although most of the existing works manually select the distillation features according to experimental experience, we hope to design a reasonable adaptive distillation position selection technique so as to select feature maps that contain more information automatically.

In further ablation experiments, we find CM loss can also significantly improve student accuracy. This is because generator is subjected to additional supervision, and tighter constraint makes the generator converge better. Moreover, if we use AT loss together with CM loss, the trained student model obtains higher accuracy of 76.71%. Student network gets best performance of 76.80% when all of these loss functions are used. Although GT loss does not significantly improve student accuracy, we find that it can greatly accelerate the convergence of models. It is also worth noting that, the generator tends to produce meaningless images in early stage, so we employ a small ratio of GT loss so that the student model subject to a strong constraint of teacher model. As the training proceeds, the synthetic images become more meaningful, and we gradually increase the ratio of GT loss. After performing the entire ablation experiments, it is shown that each of our proposed loss function is effective.

### 5 Conclusion

In this work, we propose a conditional generative data-free distillation method to train an efficient portable student network. In addition to the information captured from teacher network, preset label is introduced as auxiliary information to train the conditional generator. The trained generator produces training samples consistently so that the student model can continuously explore unseen regions and achieve better performance. To improve the distillation process, various knowledge and information are utilized to guide the distillation process, such as estimation discrepancy, preset labels, and teacher attention maps.

### Table 4 Effectiveness of different components in proposed method.

| Classes Matching loss | ✓   | ✓   | ✓   | ✓   |
|----------------------|-----|-----|-----|-----|
| Ground Truth loss    | ✓   | ✓   |     | ✓   |
| Attention Transfer loss | ✓   | ✓   | ✓   | ✓   |
| Student Accuracy     | 74.81 | 75.52 | 76.34 | 75.56 | 76.71 | 76.80 |
Furthermore, a simple training strategy is proposed to reduce the resource waste, and we design a novel evaluation metric to directly compare different distillation methods. In our work, we successfully train a portable student network without any real images and achieve state-of-the-art results on several general datasets.

In future, we will do some work on the effective representation of knowledge and the selection technique of distillation position. In addition, training a generator from scratch is computationally expensive, so further exploration is required to improve existing data-free distillation framework.

Acknowledgments. This work is supported by National Key R&D Program of China (2018YFB1308400) and Zhejiang provincial Natural Science Foundations (LY21F030018 and LQ22F030021).

References

[1] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105 (2012)

[2] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR) (2015)

[3] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 39(6), 1137–1149 (2016)

[4] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3431–3440 (2015). https://doi.org/10.1109/CVPR.2015.7298965

[5] Chen, C., Seff, A., Kornhauser, A., Xiao, J.: Deepdriving: Learning affordance for direct perception in autonomous driving. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2722–2730 (2015). https://doi.org/10.1109/ICCV.2015.312

[6] Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. nature 542(7639), 115–118 (2017)

[7] Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2Face: Real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2387–2395 (2016). https://doi.org/10.1109/CVPR.2016.262
[8] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of go with deep neural networks and tree search. nature 529(7587), 484–489 (2016)

[9] Han, S., Pool, J., Tran, J., Dally, W.: Learning both weights and connections for efficient neural network. In: Advances in Neural Information Processing Systems, vol. 28, pp. 1135–1143 (2015)

[10] He, Y., Zhang, X., Sun, J.: Channel pruning for accelerating very deep neural networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1389–1397 (2017). https://doi.org/10.1109/ICCV.2017.155

[11] Huang, Z., Wang, N.: Data-driven sparse structure selection for deep neural networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 304–320 (2018)

[12] Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. Preprint at https://arxiv.org/abs/1510.00149 (2015)

[13] Courbariaux, M., Bengio, Y., David, J.-P.: Binaryconnect: Training deep neural networks with binary weights during propagations. In: Advances in Neural Information Processing Systems, vol. 28, pp. 3123–3131 (2015)

[14] Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A.: Xnor-net: Imagenet classification using binary convolutional neural networks. In: European Conference on Computer Vision, pp. 525–542 (2016). Springer

[15] Denil, M., Shakibi, B., Dinh, L., Ranzato, M., De Freitas, N.: Predicting parameters in deep learning. In: Advances in Neural Information Processing Systems, vol. 26, pp. 2148–2156 (2013)

[16] Denton, E.L., Zaremba, W., Bruna, J., LeCun, Y., Fergus, R.: Exploiting linear structure within convolutional networks for efficient evaluation. In: Advances in Neural Information Processing Systems, vol. 27, pp. 1269–1277 (2014)

[17] Hinton, G., Vinyals, O., Dean, J., et al.: Distilling the knowledge in a neural network. Preprint at https://arxiv.org/abs/1503.02531 (2015)

[18] Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. In: International Conference on Learning Representations (ICLR) (2015)

[19] Zagoruyko, S., Komodakis, N.: Paying more attention to attention:
Improving the performance of convolutional neural networks via attention transfer. In: International Conference on Learning Representations (ICLR) (2017)

[20] Lopes, R.G., Fenu, S., Starner, T.: Data-free knowledge distillation for deep neural networks. Preprint at https://arxiv.org/abs/1710.07535 (2017)

[21] Chen, H., Wang, Y., Xu, C., Yang, Z., Liu, C., Shi, B., Xu, C., Xu, C., Tian, Q.: Data-free learning of student networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3514–3522 (2019). https://doi.org/10.1109/ICCV.2019.00361

[22] Fang, G., Song, J., Shen, C., Wang, X., Chen, D., Song, M.: Data-free adversarial distillation. Preprint at https://arxiv.org/abs/1912.11006 (2019)

[23] Luo, L., Sandler, M., Lin, Z., Zhmoginov, A., Howard, A.: Large-scale generative data-free distillation. Preprint at https://arxiv.org/abs/2012.05578 (2020)

[24] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, vol. 27, pp. 2672–2680 (2014)

[25] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: 2nd International Conference on Learning Representations (ICLR) (2014)

[26] Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., Anandkumar, A.: Born again neural networks. In: International Conference on Machine Learning (ICML), pp. 1607–1616 (2018). PMLR

[27] Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 5191–5198 (2020)

[28] Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4133–4141 (2017). https://doi.org/10.1109/CVPR.2017.754

[29] Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3967–3976 (2019). https://doi.org/10.1109/CVPR.2019.00409
[30] Bai, H., Wu, J., King, I., Lyu, M.: Few shot network compression via cross distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 3203–3210 (2020)

[31] Li, Q., Jin, S., Yan, J.: Mimicking very efficient network for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6356–6364 (2017). https://doi.org/10.1109/CVPR.2017.776

[32] Liu, Y., Shu, C., Wang, J., Shen, C.: Structured knowledge distillation for dense prediction. IEEE transactions on pattern analysis and machine intelligence (2020)

[33] Gu, J., Tresp, V.: Search for better students to learn distilled knowledge. In: ECAI 2020: 24th European Conference on Artificial Intelligence, pp. 1159–1165 (2020). IOS Press

[34] Yao, L., Pi, R., Xu, H., Zhang, W., Li, Z., Zhang, T.: Joint-detnas: Upgrade your detector with nas, pruning and dynamic distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10175–10184 (2021)

[35] Ashok, A., Rhinehart, N., Beainy, F., Kitani, K.M.: N2n learning: Network to network compression via policy gradient reinforcement learning. In: International Conference on Learning Representations (ICLR) (2018)

[36] Ma, J., Mei, Q.: Graph representation learning via multi-task knowledge distillation. In: Conference on Neural Information Processing Systems (NeurIPS) Graph Representation Learning Workshop (2019)

[37] Wang, L., Yoon, K.-J.: Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)

[38] Srinivas, S., Babu, R.V.: Data-free parameter pruning for deep neural networks. In: Proceedings of the British Machine Vision Conference 2015 (BMVC), pp. 31–13112 (2015). https://doi.org/10.5244/C.29.31

[39] Tang, J., Liu, M., Jiang, N., Cai, H., Yu, W., Zhou, J.: Data-free network pruning for model compression. In: 2021 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–5 (2021). IEEE

[40] Nagel, M., Baalen, M.v., Blankevoort, T., Welling, M.: Data-free quantization through weight equalization and bias correction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1325–1334 (2019). https://doi.org/10.1109/ICCV.2019.00141
[41] Cai, Y., Yao, Z., Dong, Z., Gholami, A., Mahoney, M.W., Keutzer, K.: Zeroq: A novel zero shot quantization framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13169–13178 (2020)

[42] Liu, Y., Zhang, W., Wang, J.: Zero-shot adversarial quantization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1512–1521 (2021). https://doi.org/10.1109/CVPR42600.2020.01318

[43] Nayak, G.K., Mopuri, K.R., Shaj, V., Radhakrishnan, V.B., Chakraborty, A.: Zero-shot knowledge distillation in deep networks. In: International Conference on Machine Learning (ICML), vol. 97, pp. 4743–4751 (2019). PMLR

[44] Yin, H., Molchanov, P., Alvarez, J.M., Li, Z., Mallya, A., Hoiem, D., Jha, N.K., Kautz, J.: Dreaming to distill: Data-free knowledge transfer via deepinversion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8715–8724 (2020). https://doi.org/10.1109/CVPR42600.2020.00874

[45] Jin, Y., Qiu, Z., Xie, G., Cai, J., Li, C., Shen, L.: Data-free knowledge distillation via adversarial. In: 2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS), pp. 447–451 (2021). IEEE

[46] Fang, G., Song, J., Wang, X., Shen, C., Wang, X., Song, M.: Contrastive Model Inversion for Data-Free Knowledge Distillation. Preprint at https://arxiv.org/abs/2105.08584 (2021)

[47] Mirza, M., Osindero, S.: Conditional generative adversarial nets. Preprint at https://arxiv.org/abs/1411.1784 (2014)

[48] Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: International Conference on Machine Learning (ICML), vol. 70, pp. 214–223 (2017). PMLR

[49] Qi, G.-J.: Loss-sensitive generative adversarial networks on lipschitz densities. International Journal of Computer Vision 128(5), 1118–1140 (2020)

[50] Lee, H.-Y., Tseng, H.-Y., Huang, J.-B., Singh, M., Yang, M.-H.: Diverse image-to-image translation via disentangled representations. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 35–51 (2018)

[51] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta,
[52] Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE transactions on image processing 26(7), 3142–3155 (2017)

[53] Buciluă, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 535–541 (2006)

[54] LeCun, Y.: The MNIST database of handwritten digits. Available at http://yann.lecun.com/exdb/mnist/ (1998)

[55] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (1998)

[56] Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)

[57] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016)

[58] Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding 106(1), 59–70 (2007)

[59] Zhao, H., Sun, X., Dong, J., Manic, M., Zhou, H., Yu, H.: Dual discriminator adversarial distillation for data-free model compression. International Journal of Machine Learning and Cybernetics, 1–18 (2021)

[60] Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: International Conference on Learning Representations (ICLR) (2016)