Accumulation Knowledge Distillation for Conditional GAN Compression

Tingwei Gao[1] Rujiao Long[1]
[1]Alibaba Group
tingwei.gtw@alibaba-inc.com, rujiao.lrj@alibaba-inc.com

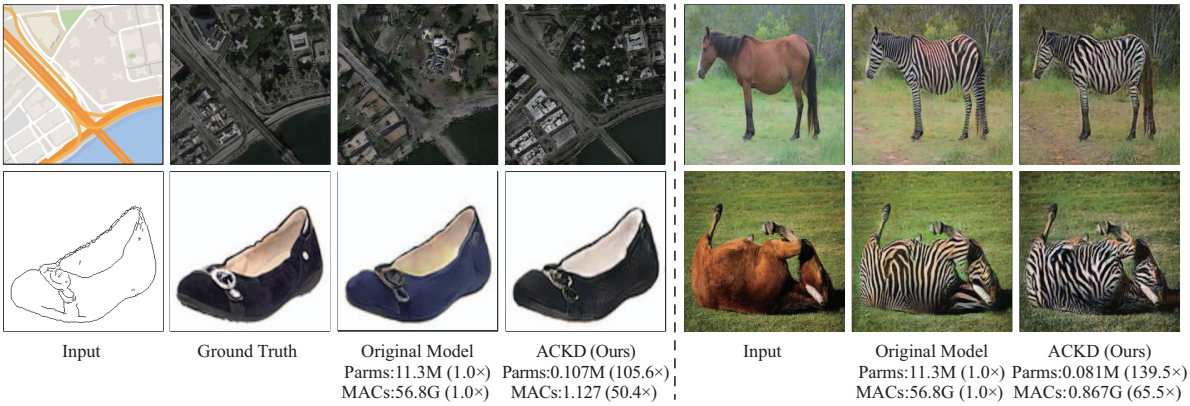

Figure 1. The compressed Pix2Pix (part a) and CycleGAN (part b) models achieved by accumulation knowledge distillation (ACKD). Parms denotes the parameters of generative model and MACs refers to the multiply-accumulate operations which are used to quantify the computation cost. ACKD largely reduces the parameters and computation while preserving the visual fidelity.

Abstract

*This paper focuses on an efficient and high-performance compression method for conditional generative adversarial networks (cGANs) from the perspective of knowledge distillation. Previous cGANs compression approaches using knowledge distillation typically transfer knowledge in a one-to-one manner, where a specific student generator layer only receives knowledge from the same depth stage in the teacher generator. Obviously, this approach fails to sufficiently explore the valuable dark knowledge embedded in the intermediate teacher generator layers. To address this issue, a novel cGANs compression method based on accumulation knowledge distillation (ACKD) is proposed. ACKD accumulates knowledge from various teacher generator stages then transfers it to the student generator. To this end, ACKD first extracts the essential knowledge from different stages and subsequently unifies them to determine their relative importance. In this manner, ACKD is capable of effectively providing hierarchical, informative and targeted knowledge to the compressed student generator. The compressed cGANs achieved by ACKD demonstrate remarkable performance surpassing other other state-of-the-art methods on three benchmarks. Furthermore, ACKD compresses parameters over **100**× and MACs over **50**×, setting new records in cGANs compression.*

1. Introduction

Conditional generative adversarial networks (cGANs) have gained significant attention due to their impressive image generation capabilities. While being successfully applied to tremendous scenarios such as image translation [1, 17, 26, 32, 39, 48] and image synthesis [2, 6, 29, 37, 45, 47], cGANs suffer from massive model size and expensive computational overhead, which significantly limit their deployment to resource-constrained platforms, especially the edge devices. This paper aims to alleviate these restrictions by proposing a novel method to compress cGANs using knowledge distillation.

Knowledge distillation for cGANs is highly non-trivial. Compared with recognition models, there is neither additional knowledge in the output layer nor a logits layer in cGANs. All the valuable knowledge for cGANs is concealed within the intermediate layers, commonly referred to as the dark knowledge. Previous approaches to compress cGANs using knowledge distillation [4, 7, 21, 30, 35] typically transfer knowledge with one-to-one connections (as

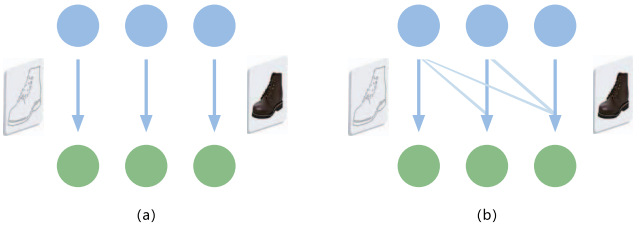

Figure 2. Different knowledge transfer approaches for cGANs. (a) is the conventional one-to-one method, while (b) is the proposed more-to-one method. The blue and green circles represent teacher and student generator structure, respectively. The lines and arrows illustrate the knowledge transfer path.

illustrated in Fig. 2 (a)), in which a certain student generator layer only gains knowledge from the corresponding teacher generator layer with the same depth. These methods fall short in fully exploring the useful dark knowledge obscured in the intermediate teacher generator layers.

To address the above issue, we propose accumulation knowledge distillation (ACKD), which transfers knowledge in a more-to-one manner (as illustrated in Fig. 2 (b)). In addition to delivering knowledge from the single teacher generator layer to the corresponding student generator layer, ACKD also accumulates knowledge from several shallow teacher generator layers in a dense-connection manner. Finer-grained structural and textural features, whose abstraction grows as the depth of neural networks increases, are necessary for the generative models to generate high-quality samples. Benefiting from the accumulation knowledge merged from down (concreteness) to top (abstraction), student generators can learn well not only the finer-grained structure, texture and detailed features in the shallow stages of teacher generators, but also features with higher abstraction in the deep stage. Furthermore, we develop Accumulation Knowledge Attention (ACK-A) as a novel method to enhance the organization of the dense accumulation knowledge. Observing that different generator layers possess distinct semantic features, ACK-A employs a two-step process to handle the accumulated knowledge. ACK-A first separates knowledge from different stages independently for spatial attention and then group them again for channel attention. This procedure is essentially ranking the knowledge importance in various layers, allowing for a more refined understanding of the knowledge dynamics. After applying ACK-A, accumulation knowledge fusion (ACK-F) fuses the reconstructed accumulation knowledge and the deep stage feature.

Inspired by [16, 46], attempts are made to improve the performance of the teacher model by self-distillation using ACKD. Such approach involves the teacher serving as its own teacher. A great teacher will directly contribute to good results for knowledge distillation. However, as experimentally verified by [8, 25], when the structure of teacher and that of student is vastly different, the knowledge distillation performance will be reduced. Therefore, based on self-distillation, the capability of the teacher generator can be enhanced without enlarging the structural gap between the teacher and student generators.

The present method is evaluated on three benchmarks. In the experiments, ACKD demonstrates the ability to preserve visual fidelity while utilizing significantly fewer parameters and MACs (as shown in Fig. 1). Impressively, even with compression ratios as high as **105.6**$\times$ for paramters and **50.4**$\times$ for MACs, or compression ratios of **139.5**$\times$ for parameters and **65.5**$\times$ for MACs, ACKD is capable of achieving state-of-the-art results. The experimental results demonstrate that a compressed student generator can be both small and effective when guided by efficient and reasonable knowledge from the intermediate teacher generator layers in the training process.

The major contributions of this work can be summarized as follows: (1) This paper introduces and highlights the significance of accumulating knowledge for the knowledge distillation of cGANs. (2) This work proposes ACK-A, a novel framework that effectively organizes the accumulation knowledge in a reasonable, appropriate, and targeted manner. The efficiency of ACK-A is demonstrated through a sufficient amount of ablation study. (3) The proposed ACKD approach achieves state-of-the-art performance in cGAN compression, surpassing previous methods by a significant margin.

2. Related Work

2.1. cGANs and cGANs Compression

cGANs. A significant branch of GAN [10] is conditional GAN (cGAN), the core idea of which is to control images generated by condition, rather than a random manner. cGAN consists of a generator G and a discriminator D. In the training process of cGANs, G generates realistic images to deceive D, while D distinguishes the images generated by G from the real ones. cGANs have been extensively adopted in image-to-image tasks in recent years, with Pix2Pix [17] and CycleGAN [48] as two of the more representative cGAN models. Pix2Pix and CycleGAN can both be applied to image translation, but the difference is that the former is used to process data in the paired form, while the latter is for data in the unpaired form. Recently, there has been an emergence of cGANs [2, 24, 26, 38, 39] that can generate photo-realistic images, however such cGANs require a considerably high number of parameters and a significantly large amount of computation. Hence, cGANs compression is of high practical significance.

cGANs Compression. Recently, a number of notable research efforts focusing on cGANs compression to reduce the number of parameters and computation of cGANs

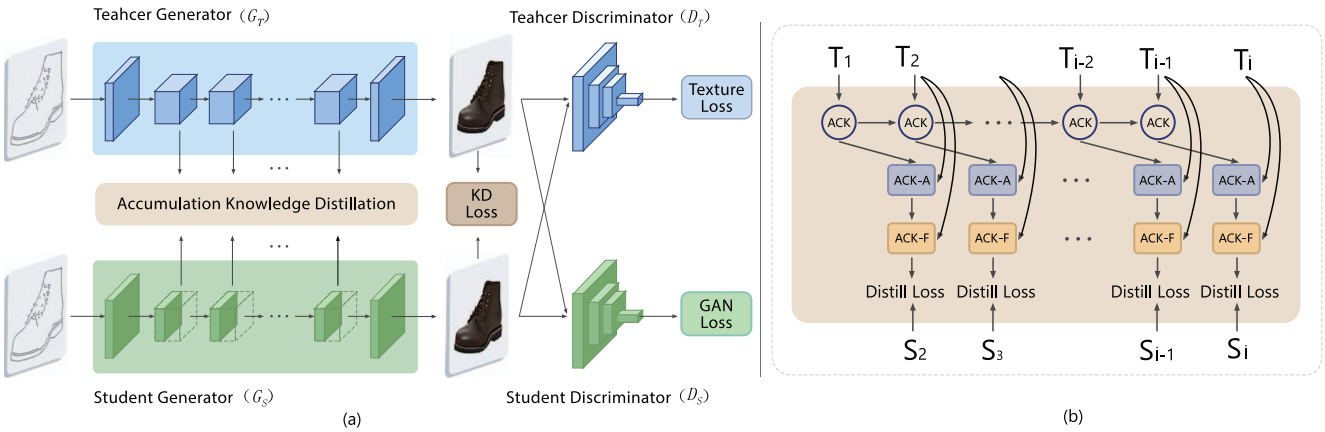

Figure 3. The overview of the proposed cGANs compression method based on accumulation knowledge distillation. (a) shows the pipeline and (b) illustrates the framework of accumulation knowledge distillation. ACK denotes accumulation knowledge, ACK-A denotes the accumulation knowledge attention and ACK-F denotes the accumulation knowledge fusion.

have been proposed, thereby advancing the application of cGANs. The more representative work is [21], which standardizes the experimental setup for subsequent related work, including the benchmark datasets and experimental metrics. An online knowledge distillation with multi-granularity loss and two teacher generators was proposed by [30], which achieves impressive results by compressing parameters over 80 times. Regarding [4], a GAN compression method was designed using knowledge distillation, and a triplet loss was also employed to train the discriminator. Fu *et al.* [7] used neural architecture search and knowledge distillation in combination and implemented a GAN compression method called Auto-GAN. Li *et al.* [22] developed a GAN compression algorithm based on differentiable mask and co-attention. Jin *et al.* [19] introduced a new generator structure, and then implemented a GAN compression solution via the pure algorithm. Wang *et al.* [35] proposed a GAN compression framework by combining knowledge distillation, quantization training, and channel pruning.

2.2. Knowledge Distillation

Knowledge Distillation. Knowledge distillation is an essential model compression method, and its function is to distill and extract the knowledge contained in an already trained model named teacher into another model named student. Using the output of the logits layer of teacher to guide the training of student was proposed by [15] in 2015. Romero *et al.* [31] proposed FitNets, which marked the beginning of feature-based knowledge distillation. The majority of existing feature-based knowledge distillation methods [13, 27, 28, 34] have only adopted a one-to-one manner, that is, the training of the corresponding stage of the student is instructed with the knowledge of the teacher network at the same stage. Some works [42, 44] attempt to use multi-layers in the same stage.

Cross-stage Distillation. The first cross-stage based knowledge distillation is knowledge review proposed by [5], which is excellent work for knowledge distillation. However, this method involving more layers of knowledge is not the same as the proposed ACKD in more dense manner, and does not deal with multiple layers of knowledge in a more fine-grained and targeted manner.

Self-Distillation. Self-distillation was first proposed by [46], in which the teacher and student have the same network structure. Hou *et al.* [16] used self-distillation for lane detection and Hahn *et al.* [12] explored the capacity of self-distillation for natural language processing. Our paper is the first to apply and validate self-distillation for the generative models.

3. Method

3.1. Framework Overview

The framework is illustrated in Fig. 3. G_T and D_T denote the generator and discriminator of teacher, respectively, while G_S and D_S represent corresponding student. G_T and G_S have the same depth and are divided into different stages using the same method. ($T_1, T_2, \ldots, T_m$) and ($S_1, S_2, \ldots, S_m$) represent the different stages of G_T and G_S. As shown in Fig. 3 (b), for a certain stage of G_T, knowledge in the shallow stages before it is the accumulation knowledge. G_S receives the knowledge from both the same stage of G_T and the corresponding accumulation knowledge. Under the guidance of the deep-stage knowledge of G_T, the accumulation knowledge is effectively and purposefully selected, organized and integrated through ACK-A, and then fused with the deep-stage knowledge through ACK-F. The knowledge after fusion is used to instruct G_S on the training of the corresponding stage.

For the supervision of losses, as shown in Fig. 3 (a), G_S

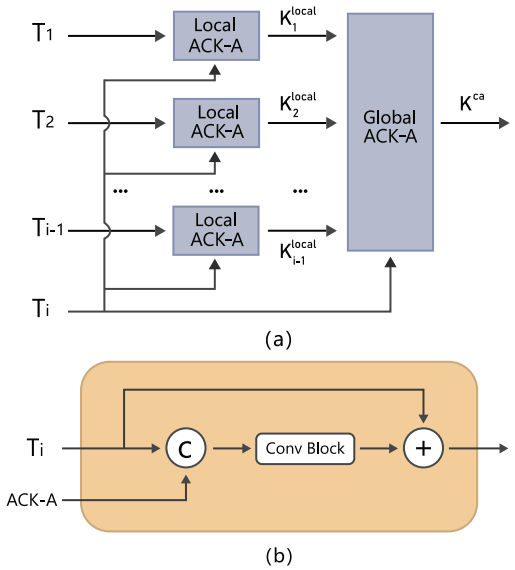

Figure 4. The illustration of modules in ACKD. (a) is the pipeline of ACK-A, (b) is the structure of ACK-F. Ⓒ denotes the channel-wise concatenation, ⊕ denotes the element-wise sum, and conv block denotes several convolution operations.

is only allowed to view G_T instead of both G_T and the ground truth. Meanwhile, given that D_T holds a significant amount of related prior knowledge that can be used to guide the training of G_S, this work takes full advantage of it. The learning of Texture loss of G_S is supervised by D_T to better learning of the texture feature from G_T.

3.2. Accumulation Knowledge Distillation

Fig. 4 (a) illustrates the pipeline of ACK-A. For the purpose of effective and targeted organization of accumulation knowledge, the process of ACK-A is conducted under the guidance of its corresponding deep-stage knowledge. Firstly, knowledge of different stages is guided by deep-stage knowledge to separately and independently conduct spatial attention. Such a process is referred to as local ACK-A, which marks the key points of knowledge in different stages of accumulation knowledge with different perspectives. Subsequently, all the outputs of local ACK-A are concatenated together and jointly conducted channel attention under the guidance of knowledge in deep stage. Such process is global ACK-A, which ranks the importance of knowledge of all stages together.

The following is the process of ACK-A in detail. For the sake of brevity, the reshaping feature map in related operations is omitted. Given $T_i \in \mathbb{R}^{C\times H\times W}$ (C, H and W denote the channel number, height and width of feature map, respectively), and T_j is set to represent one of the shallow stages of T_i. For the purpose of obtaining the spatial attention map $\mathbf{A}_j^{local} \in \mathbb{R}^{1\times H\times W}$ for T_j under the guidance of T_i, query vector $\mathbf{q}_{i\to j}^{local}$ from T_i, and key vector

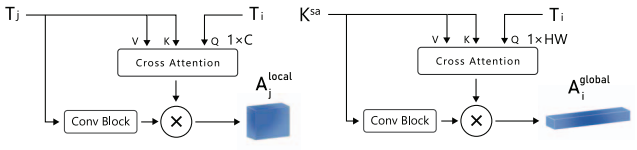

Figure 5. The illustration of ACK-A. The left part is the diagram of local ACK-A generating spatial attention map, the right part is the diagram of global ACK-A generating channel attention map. ⊗ denotes the matrix multiplication operation.

$\mathbf{k}_j$ and value vector $\mathbf{q}_j$ from T_j are obtained as follows:

$$\begin{aligned} \mathbf{q}_{i\to j}^{local} &= \mathbf{f}_{i\to j}((\phi^{HW}(\mathbf{W}_i T_i))) \\ \mathbf{k}_j &= \mathbf{f}_j^k(\phi^{HW}(\mathbf{W}_j^{kv} T_j)) \\ \mathbf{v}_j &= \mathbf{f}_j^v(\phi^{HW}(\mathbf{W}_j^{kv} T_j)), \end{aligned} \tag{1}$$

where ϕ^{HW} represents the global average pooling layer which is used for obtaining global spatial semantic information, $\mathbf{W}_i$ and $\mathbf{W}_j^{kv}$ represent the 1×1 convolution layers and $\mathbf{f}_{i\to j}$, $\mathbf{f}_j^k$ and $\mathbf{f}_j^v$ represent the full connection layers which are used to obtain the targetd vector..

Afterwards, as shown in Fig. 5, the cross-attention module [41] is adopted to establish the relationship between T_i and T_j, outputting it as dynamic weight. Subsequently, the spatial attention map $\mathbf{A}_\mathbf{j}^{\mathbf{local}}$ which indicates the importance of each pixel in guiding student learning for T_j, and $\mathbf{K}_\mathbf{j}^{\mathbf{local}}$ can be obtained by means of the following:

$$\begin{aligned} \mathbf{A}_\mathbf{j}^{\mathbf{local}} &= \sigma(\mathbf{W}_j T_j \mathbf{A}(\mathbf{q}_{i\to j}^{local}, \mathbf{k}_j, \mathbf{v}_j)) \\ \mathbf{K}_\mathbf{j}^{\mathbf{local}} &= \mathbf{A}_\mathbf{j}^{\mathbf{local}} T_j, \end{aligned} \tag{2}$$

where $\mathbf{W}_j$ is a 1×1 convolution layer, σ is the sigmoid function, $\mathbf{A}$ is the cross attention module [41].

When the local ACK-A is completed for all stages of accumulation knowledge, their outputs are concatenated and denoted as K_i^{sa}. Next, global query vector $\mathbf{q}_i^{global}$ from T_i, key vector $\mathbf{k}_i^{sa}$ and value vector $\mathbf{v}_i^{sa}$ from K_i^{sa} are obtained by means of the following:

$$\begin{aligned} \mathbf{q}_i^{global} &= \mathbf{f}_\mathbf{i}((\phi^C(\mathbf{W}_i T_i))) \\ \mathbf{k}_i^{sa} &= \mathbf{f}_{sa-i}^k(\phi^C(\mathbf{W}^{kv} K_i^{sa})) \\ \mathbf{v}_i^{sa} &= \mathbf{f}_{sa-i}^v(\phi^C(\mathbf{W}^{kv} K_i^{sa})), \end{aligned} \tag{3}$$

where ϕ^C represents the channel-wise pooling which is used for obtaining global channel semantic information, $\mathbf{f_i}$, $\mathbf{f}_{sa-i}^k$ and $\mathbf{f}_{sa-i}^k$ represent the full connection layers and $\mathbf{W}_i$ and $\mathbf{W}^{kv}$ represent the 1×1 convolution layers.

Then, the channel attention map $\mathbf{A}_\mathbf{i}^{\mathbf{global}} \in \mathbb{R}^{C\times1\times1}$ as shown below and K^{ca} can be obtained:

$$\begin{aligned} \mathbf{A}_\mathbf{i}^{\mathbf{global}} &= softmax(\mathbf{W}_i^{sa} K_i^{sa} \mathbf{A}(\mathbf{q}_i^{global}, \mathbf{k}_i^{sa}, \mathbf{v}_i^{sa})) \\ K^{ca} &= \mathbf{A}_\mathbf{i}^{\mathbf{global}} K^{sa}, \end{aligned} \tag{4}$$

where W_i^{sa} denotes a 1×1 convolution layer.

After the aforementioned operations, targeted reorganization of accumulation knowledge can be obtained, which is then fused with T_i through ACK-F (as shown in Fig. 4 (b)). As a result, the available ACKD loss is:

$$\mathcal{L}_{ackd} = \sum^{m} ||\mathbf{Fusion}(T_i, K_i^{ca}), \mathbf{W}_S(S_i))||_1, \tag{5}$$

where $\mathbf{W}_S$ represents a 1×1 convolution layer, used to deal with the different shapes of features of G_T and G_S. $||||_1$ denotes L1 distance, **Fusion** represents ACK-F.

In regard to self-distillation, student generator equips exactly the same network structure as teacher generator. T_i' is used to represent the corresponding stage in the student generator, and the following loss can be obtained:

$$\mathcal{L}_{ackd-self} = \sum^{m} ||\mathbf{Fusion}(T_i, K_i^{ca}), T_i'||_1. \tag{6}$$

3.3. Training objectives

Different from knowledge distillation for tasks such as image recognition [18, 25] and object detection [3, 11], which provide supervision for student model training jointly using both the intermediate features of the teacher and the ground truth, for the compressed student generator, all the supervision on the losses comes from G_T. This study is not the first one to adopt ground truth free setting for cGANs compression. [30] also uses this setting and gives the explanation that such method can reduce the difficulty of training. However, [30] dose not give detailed reasons and ablation experiments. Nonetheless, this study wants to give a point of view. Compared with tasks such as image recognition and object detection, the answer of cGANs is not unique. It is a correct output result just as long as the output image conforms to the cognitive common sense of human. However, every detail or texture of the output image of G_T remaining the same as the ground truth cannot be guaranteed. Therefore, when the parameters of G_S is relatively small, as the learning capacity is limited, in which case if there is a gap between the supervision from knowledge of the intermediate layers of teacher and ground truth images, then the training of G_S will be affected.

This work adds Texture loss and SSIM loss to the original loss method in [17, 48]. The performance differences among various models of cGAN mainly lies in image texture, which is attributed to the fact that cGAN typically generates or modifies texture based on given information. Therefore, it is essential to emphasize the texture features for cGAN compression. SSIM is capable of making the generated images better conform to human visual perception and present more details.

Given input image X, and $G_T(X)$ and $G_S(X)$ are denoted as g_t and g_s. The SSIM loss [40] is as follow:

$$\mathcal{L}_{ssim} = ||SSIM(g_t) - SSIM(g_s)||_1. \tag{7}$$

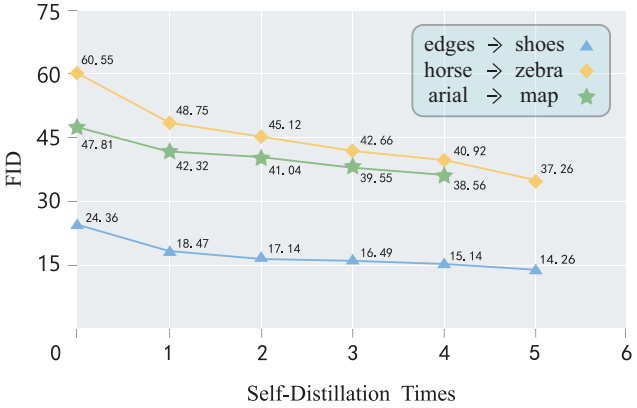

Figure 6. The illustration of the performance of self-distillation using ACKD. When the value of abscissa is 0, the FID is the performance of the original teacher generator.

The reconstruction loss provides pixel-wise supervision.

$$\mathcal{L}_{recon} = ||g_t - g_s||_1. \tag{8}$$

The total KD loss is as folllow:

$$\mathcal{L}_{KD} = \lambda_{recon}\mathcal{L}_{recon} + \lambda_{ssim}\mathcal{L}_{ssim}, \tag{9}$$

where λ_{recon} and λ_{ssim} are applied to balance losses.

The Texture loss provides supervision on texture feature.

$$\mathcal{L}_{texture} = \sum^{k} ||\Phi(D_T^i(g_t)) - \Phi(D_T^i(g_s))||_1, \tag{10}$$

where D_T^i denotes the ith layer activation in D_T, Φ denotes the $gram\ matrix$.

The adversarial loss is also applied.

$$\begin{aligned}\mathcal{L}_{gan} =& \mathbb{E}_X[log(1 - D_S(X, G_S(X)))] \\ &+ \mathbb{E}_{X,G_T(X)}[log(D_S(X, G_T(X)))].\end{aligned} \tag{11}$$

The overall training objectives are by follow:

$$\mathcal{L} = \lambda_{ack}\mathcal{L}_{ackd} + \mathcal{L}_{KD} + \lambda_{tex}\mathcal{L}_{texture} + \mathcal{L}_{gan}, \tag{12}$$

where λ_{ack} and λ_{tex} are adopted as weights for losses.

4. Experiments

4.1. Experimental Settings

Models. Following the previous cGANs compression work, the compression experiments are performed on Pix2Pix [17] and CycleGAN [48]. The network structure is the same as that of [21, 22, 30].

Datasets Description. Three of the most commonly utilized datasets for cGANs compression research are adopted for the experiments, which facilitates the acquisition of baseline models for comparison. (1) The horse→zebra [48],

Table 1. Comparison with other cGANs compression methods on the horse→zebra. CR represents compression ratio.

Method	MACs (CR)	Parameters (CR)	FID ↓
Original [48]	56.8G (1.0×)	11.3M (1.0×)	61.53
Co-Evol [20]	13.4G (4.2×)	_	96.15
GAN-Slim [35]	11.34G (5.0×)	_	86.09
AGD [7]	6.39G (8.9×)	_	83.60
GAN Comp [21]	2.67G (21.3×)	0.34M (33.2×)	64.95
DMAD [22]	2.41G (23.6×)	0.28M (40.4×)	62.96
CF-GAN [36]	2.65G (21.4×)	_	62.31
Gong *et al.* [9]	1.57G (36.2×)	0.22M (51.4×)	60.49
CAT [19]	2.55G (22.3×)	0.43M(26.3×)	60.18
GCC [23]	2.40G (23.7×)	_	59.31
OMGD [30]	1.408G (40.3×)	0.137M (82.5×)	51.92
Ours-a	2.904G (19.6×)	0.296M (38.2×)	38.69
Ours-b	1.408G (40.3×)	0.137M(82.5×)	39.67
Ours-c	1.127G (50.4×)	0.107M (105.6×)	40.62
Ours-d	0.867G (65.5×)	0.081M (139.5×)	45.24

Table 2. Comparison with other cGANs compression methods on the edges→shoes. CR represents compression ratio.

Method	MACs (CR)	Parameters (CR)	FID ↓
Original [17]	56.8G (1.0×)	11.3M (1.0×)	24.18
GAN Comp [21]	4.81G (11.8×)	0.70M (16.1×)	26.60
GF-GAN [36]	4.77G (11.9×)	-	24.13
DMAD [22]	4.30G (13.2×)	0.54M (20.9×)	24.08
OMGD-b [30]	1.408G (40.3×)	0.137M (82.5×)	25.88
OMGD-a [30]	2.904G (19.6×)	0.296M (38.2×)	21.41
Ours-a	2.904G (19.6×)	0.296M (38.2×)	16.46
Ours-b	1.408G (40.3×)	0.137M (82.5×)	22.10
Ours-c	1.127G (50.4×)	0.107M (105.6×)	23.24
Ours-d	0.867G (65.5×)	0.081M (139.5×)	26.61

Table 3. Comparison with other cGANs compression methods on the aerial→map. CR represents compression ratio.

Method	MACs (CR)	Parameters (CR)	FID ↓
Original [17]	56.8G (1.0×)	11.3M (1.0×)	47.76
GAN Comp [21]	4.68G (12.1×)	0.75M (15.1×)	48.02
Gong *et al.* [9]	4.56G (12.5×)	0.51M (22.2×)	47.32
CF-GAN [36]	4.50G (12.6×)	-	46.15
CAT [19]	4.59G (12.4×)	0.54M(20.9×)	44.96
Ours-a	2.904G (19.6×)	0.296M (38.2×)	41.35
Ours-b	1.408G (40.3×)	0.137M (82.5×)	43.28
Ours-c	1.127G (50.4×)	0.107M (105.6×)	44.82
Ours-d	0.867G (65.5×)	0.081M (139.5×)	49.94

the input is a horse image and the output is a zebra image. (2) The edges→shoes [43], the input is a sketch of the structure of a shoe, while the output is an image of the actual shoe. (3) The map→aerial [17], the input is an map image, the output is an aerial photo. CycleGAN compression experiments are performed on the horse→zebra, while Pix2Pix compression experiments are conducted on the other datasets. To ensure a fair comparison, the data setting is identical to [19, 21, 22, 30].

Performance Metric. For obtaining baseline metrics to evaluate ACKD, we follow the previous cGAN compression work [19, 21, 22, 23, 30, 36] to apply the Fréchet Inception Distance (FID) [14] as the performance metric for the three datasets. FID estimates the distribution between images using pre-trained Inception [33] and a lower FID value indicates better generator performance.

Implementation Details. The experiments are conducted using Pytorch. For the edges→shoes, the batch size is set to 4, while for the other two datasets, the batch size is set to 1. Consistent with previous work, the input images are resized to 256×256. Adam is applied as the optimizer. As with [19, 21], the student discriminator inherits the pre-trained weights of the teacher discriminator.

4.2. Experimental Results

4.2.1 Quantitative Results

The self-distillation experiments using ACKD are conducted, and the original teacher generators are the pre-trained models provided by [21]. After the improvement, the teacher generator may potentially further improve itself by self-distillation again. Based on the teacher model obtained by last self-distillation, the self-distillation experiment is repeated until the performance of the teacher generator is no longer improved or the improvement is small enough to be neglected. Fig. 6 illustrates the results on three datasets. The performance of the teacher generator has been remarkably improved after 4 or 5 times of self-distillation using ACKD.

Based on the improved teacher generator by self-distillation, the experiments of generator compression are embarked on. This study trains the models with varying compression degrees on the three datasets. The evaluation results for the horse→zebra are presented in Tab. 1. It is noticeable that our models outperform all previous cGAN compression methods by a considerable margin. When ACKD compresses parameters 139.5× and MACs 65.5×, our model still maintain the superior performance. Tab. 2 shows the experiment results on the edges→shoes. Under the same compression degree, it is evident that ACKD has better performance. Additionally, competitive results can still be obtained when the parameters are compressed to over 100× and the MACs are compressed to over 50×. According to Tab. 3, in comparison to the alternative compression methods, ACKD demonstrates clear advantages on the map→aerial. With 105.6× parameters compression and 50.4× MACs compression, FID achieved by ACKD still

Figure 7. Qualitative comparisons with other methods. The left upper part are the compression results on the map→aerial, the left lower part are experimental results on the edges→shoes and the right part are the visualization results on the horse→zebra. With the same or fewer parameters, the images achieve by ACKD have better visual fidelity.

remains superior to other cGANs compression methods.

4.2.2 Ablation Study

In order to verify the effectiveness of each component in ACKD, the ablation study is conducted on edges→shoes for both teacher generator self-distillation and student generator compression, and various setting experiments are carried out as follows: (a) w/o ACK-A, remove ACK-A and apply convolution layers to fuse the accumulation knowledge. (b) w/o Local ACK-A, remove local ACK-A. (c) w/o Global ACK-A, remove global ACK-A. (d) Local CA, knowledge of different stages is conducted channel attention independently. (e) Global SA, knowledge of different stages is unified and conducted spatial attention jointly. (f) w/o guidance, ACK-A is performed without the guidance of the corresponding deep-stage knowledge. (g) C-KD, the conventional feature-based knowledge distillation method, namely the one-to-one method, is applied. (h) Less stages, the number of stages of accumulation knowledge is reduced, involving only adjacent stage. (i) only w/ $\mathcal{L}_{ori}$, the student generator is trained with $\mathcal{L}_{ackd}$ and the original loss method in [17]. (j) w/o $\mathcal{L}_{ssim}$, remove SSIM loss. (k) w/o $\mathcal{L}_{texture}$, remove Texture loss. (l) w/o gt-free, besides $\mathcal{L}_{ackd}$, all supervision on other losses comes from the ground truth.

According to the experimental results in Tab. 4, the following observations and analysis can be drawn. The performance of the model is degraded to different degrees when ACK-A or some of its modules are removed or

Table 4. The ablation study about the effective of ACK-A and accumulation knowledge for self-distillation and compressing parameters 82.5× on edges→shoes.

Tag	Method	Self-Distillation	Compression
(a)	w/o ACK-A	20.13	24.78
(b)	w/o Local ACK-A	19.23	24.20
(c)	w/o Global ACK-A	19.69	24.34
(d)	Local CA	19.81	23.79
(e)	Global SA	18.99	23.83
(f)	w/o guidance	19.39	23.89
(g)	C-KD	20.85	25.81
(h)	Less stages	19.60	24.05
	ours	18.47	22.10

Table 5. The ablation study about the loss methods for self-distillation and compressing parameters 82.5× on edges→shoes.

Tag	Method	Self-Distillation	Compression
(i)	only w/ $\mathcal{L}_{ori}$	19.32	23.15
(j)	w/o $\mathcal{L}_{ssim}$	18.90	22.84
(k)	w/o $\mathcal{L}_{texture}$	19.02	22.96
	ours	18.47	22.10

Figure 8. Qualitative comparisons for self-distillation. The second column is the original teacher generator, while the third column is the teacher generator obtained by self-distillation using ACKD.

Table 6. The ablation study about the gt-free for compressing parameters 82.5× on edges→shoes.

Tag	Method	Compression
(l)	w/o gt-free	47.38
	ours	22.10

the implementation of some modules is modified. It demonstrates that although accumulation knowledge is able to help model training, a well-organized and appropriate implementation is crucial for better utilization. When the guidance of corresponding deep-stage knowledge is removed, the experimental results decline as a consequence of the absence of more targeted attention operations. The experimental results degrades when the number of stages decreases, suggesting that more accumulation knowledge can be beneficial to the training of the model. Benefiting from the teacher with improved performance, even the compressed generator obtained by conventional knowledge distillation method can still achieve a competitive result. Nevertheless, compared to ACKD, the conventional knowledge distillation shows significantly less improvement.

Tab. 5 presents the ablation study conducted on loss functions. Notably, even when the model is trained using only $\mathcal{L}_{ackd}$ and the original loss in [17], there is still a significant improvement for the performance of model, thus demonstrating the effectiveness of ACKD for cGANs. Additionally, both $\mathcal{L}_{ssim}$ and $\mathcal{L}_{texture}$ are capable of further improving the performance, thereby it is beneficial to incorporate these loss functions in the training process.

The experimental results for gt-free are reported in Tab. 6. It is noteworthy that the performance of student generator exhibits a significant decline when the supervisions of some losses switch from teacher generator to ground truth. This observation indicates that when the number of parameters of generator is small and its learning performance is relatively low, the existence of a certain gap in the supervision of different losses will exert a considerable influence on the training of compressed generators.

4.2.3 Qualitative Results

In each of the three datasets, the model with the most pioneering advantages in our approache is selected to conduct the qualitative analysis with other methods.Fig. 7 shows that, on the map→aerial dataset, the images generated by ACKD have superior texture effect of road, rivers and buildings compared to other models, especially in the river regions, where ACKD generates more realistic and clearer rivers. On the edges→shoes dataset, the leather shoe images achieved by ACKD have better leather material gloss and fewer artifacts than other images. On the horse→zebra dataset, ACKD generates images with better performance on the black and white stripe texture of zebra skin compared to other methods, and other methods preserve more of the original horse skin color from the input images.

Fig. 8 illustrates the qualitative comparisons of the teacher generator after performance improvement with self-distillation using ACKD. It is evident that the images generated by our teacher generator have a tangible improvement in the quality, including better texture effects and less artifacts than the original teacher.

5. Conclusion

In this paper, a novel cGANs compression method is introduced. The proposed ACKD not only facilitates the training of the compressed generator but also improves the performance of the teacher generator through self-distillation. Experimental results demonstrate that efficient and reasonable knowledge extraction from the intermediate layers of the teacher generator can lead to promising results in guiding the training of the compressed generator.

References

[1] Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. Artflow: Unbiased image style transfer via reversible neural flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 862–871, 2021.

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[3] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017.

[4] Hanting Chen, Yunhe Wang, Han Shu, Changyuan Wen, Chunjing Xu, Boxin Shi, Chao Xu, and Chang Xu. Distilling portable generative adversarial networks for image translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3585–3592, 2020.

[5] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5008–5017, 2021.

[6] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14131–14140, 2021.

[7] Yonggan Fu, Wuyang Chen, Haotao Wang, Haoran Li, Yingyan Lin, and Zhangyang Wang. Autogan-distiller: Searching to compress generative adversarial networks. *arXiv preprint arXiv:2006.08198*, 2020.

[8] Mengya Gao, Yujun Wang, and Liang Wan. Residual error based knowledge distillation. *Neurocomputing*, 433:154–161, 2021.

[9] Luqi Gong, Chao Li, Hailong Hong, Hui Zhu, Tangwen Qian, and Yongjun Xu. Towards compressing efficient generative adversarial networks for image translation via pruning and distilling. In *International Conference on Artificial Neural Networks*, pages 637–647. Springer, 2021.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. 2014.

[11] Jianyuan Guo, Kai Han, Yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2154–2164, 2021.

[12] Sangchul Hahn and Heeyoul Choi. Self-knowledge distillation in natural language processing. *arXiv preprint arXiv:1908.01851*, 2019.

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

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[15] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[16] Yuenan Hou, Zheng Ma, Chunxiao Liu, and Chen Change Loy. Learning lightweight lane detection cnns by self attention distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1013–1021, 2019.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[18] Mingi Ji, Byeongho Heo, and Sungrae Park. Show, attend and distill: Knowledge distillation via attention-based feature matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7945–7952, 2021.

[19] Qing Jin, Jian Ren, Oliver J Woodford, Jiazhuo Wang, Geng Yuan, Yanzhi Wang, and Sergey Tulyakov. Teachers do more than teach: Compressing image-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13600–13611, 2021.

[20] Qing Jin, Jian Ren, Oliver J Woodford, Jiazhuo Wang, Geng Yuan, Yanzhi Wang, and Sergey Tulyakov. Teachers do more than teach: Compressing image-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13600–13611, 2021.

[21] Muyang Li, Ji Lin, Yaoyao Ding, Zhijian Liu, Jun-Yan Zhu, and Song Han. Gan compression: Efficient architectures for interactive conditional gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5284–5294, 2020.

[22] Shaojie Li, Mingbao Lin, Yan Wang, Chao Fei, Ling Shao, and Rongrong Ji. Learning efficient gans for image translation via differentiable masks and co-attention distillation. *IEEE Transactions on Multimedia*, 2022.

[23] Shaojie Li, Jie Wu, Xuefeng Xiao, Fei Chao, Xudong Mao, and Rongrong Ji. Revisiting discriminator in gan compression: A generator-discriminator cooperative compression scheme. *Advances in Neural Information Processing Systems*, 34:28560–28572, 2021.

[24] Jie Liang, Hui Zeng, and Lei Zhang. High-resolution photo-realistic image translation in real-time: A laplacian pyramid translation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9392–9400, 2021.

[25] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198, 2020.

[26] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference*

on computer vision and pattern recognition, pages 2337–2346, 2019.

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

[28] Nikolaos Passalis, Maria Tzelepi, and Anastasios Tefas. Probabilistic knowledge transfer for lightweight deep representation learning. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2030–2039, 2020.

[29] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International conference on machine learning*, pages 1060–1069. PMLR, 2016.

[30] Yuxi Ren, Jie Wu, Xuefeng Xiao, and Jianchao Yang. Online multi-granularity distillation for gan compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6793–6803, 2021.

[31] 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.

[32] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409, 2017.

[33] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[34] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

[35] Haotao Wang, Shupeng Gui, Haichuan Yang, Ji Liu, and Zhangyang Wang. Gan slimming: All-in-one gan compression by a unified optimization framework. In *European Conference on Computer Vision*, pages 54–73. Springer, 2020.

[36] Jiahao Wang, Han Shu, Weihao Xia, Yujiu Yang, and Yunhe Wang. Coarse-to-fine searching for efficient generative adversarial networks. *arXiv preprint arXiv:2104.09223*, 2021.

[37] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[38] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[39] Xinrui Wang and Jinze Yu. Learning to cartoonize using white-box cartoon representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8090–8099, 2020.

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[41] Xi Wei, Tianzhu Zhang, Yan Li, Yongdong Zhang, and Feng Wu. Multi-modality cross attention network for image and sentence matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10941–10950, 2020.

[42] 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.

[43] Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 192–199, 2014.

[44] 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.

[45] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018.

[46] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3713–3722, 2019.

[47] Pengze Zhang, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Exploring dual-task correlation for pose guided person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7713–7722, 2022.

[48] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.