Online Multi-Granularity Distillation for GAN Compression

Yuxi Ren*   Jie Wu*   Xuefeng Xiao   Jianchao Yang

ByteDance Inc.

{renyuxi.20190622, wujie.10, xiaoxuefeng.ailab, yangjianchao}@bytedance.com
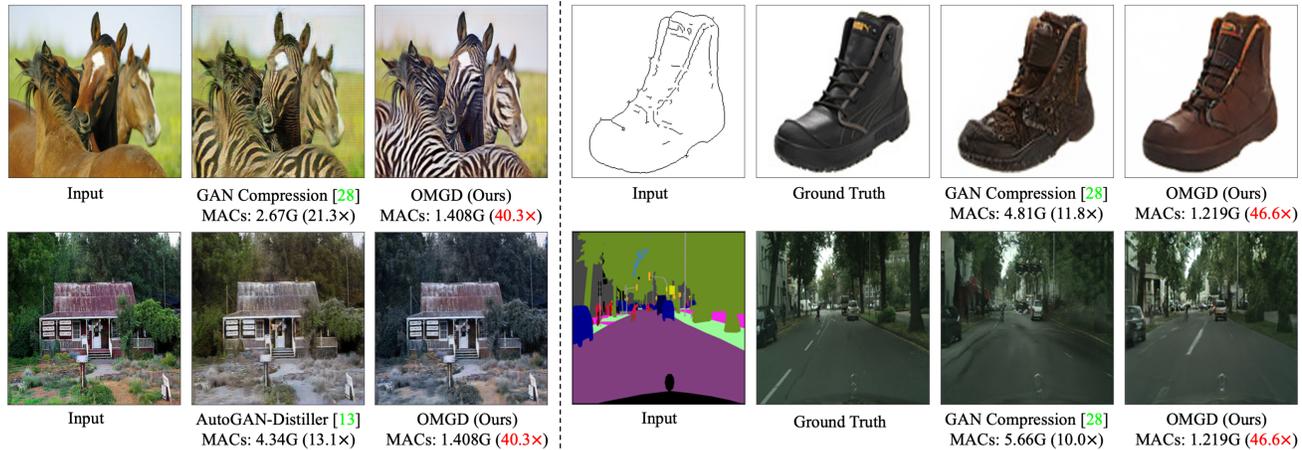
Figure 1: We introduce Online Multi-Granularity Distillation (OMGD) scheme for compressing conditional GANs. OMGD reduces the computation of Pix2Pix and CycleGAN by **40.3-46.6×** while preserving the visual fidelity.

Abstract

Generative Adversarial Networks (GANs) have witnessed prevailing success in yielding outstanding images, however, they are burdensome to deploy on resource-constrained devices due to ponderous computational costs and hulking memory usage. Although recent efforts on compressing GANs have acquired remarkable results, they still exist potential model redundancies and can be further compressed. To solve this issue, we propose a novel online multi-granularity distillation (OMGD) scheme to obtain lightweight GANs, which contributes to generating high-fidelity images with low computational demands. We offer the first attempt to popularize single-stage online distillation for GAN-oriented compression, where the progressively promoted teacher generator helps to refine the discriminator-free based student generator. Complementary teacher generators and network layers provide comprehensive and multi-granularity concepts to enhance visual fidelity from diverse dimensions. Experimental results on four benchmark datasets demonstrate that OMGD successes to compress 40× MACs and 82.5× parameters on Pix2Pix and CycleGAN, without loss of image qual-ity. It reveals that OMGD provides a feasible solution for the deployment of real-time image translation on resource-constrained devices. Our code and models are made public at: https://github.com/bytedance/OMGD

1. Introduction

Recently, Generative Adversarial Networks (GANs) [16] have achieved prominent results in diversified visual applications, such as image synthesis [39, 40, 34, 64, 5] and image-to-image translation [23, 66, 10, 26, 11, 47]. Albeit with varying degrees of progress, most of its recent successes [23, 66, 47, 64, 10, 5] are involved in huge resource demands. It is arduous to popularize such models that require tremendous computational costs, which becomes a critical bottleneck as this model is deployed on resource-constrained mobile phones or other lightweight IoT devices [21, 50, 30, 9]. To alleviate such expensive and unwieldy computational costs, GAN compression becomes a newly-raised and crucial task. A great deal of mainstream model compression techniques [29, 28, 56, 38, 57, 33] are employed to learn efficient GAN, including knowledge distillation [30, 1, 49, 8, 13, 15, 31, 20, 9, 24], channel pruning [30, 31, 49] and nerual architecture search [15, 30, 31].

[1]These authors contributed equally to this work.
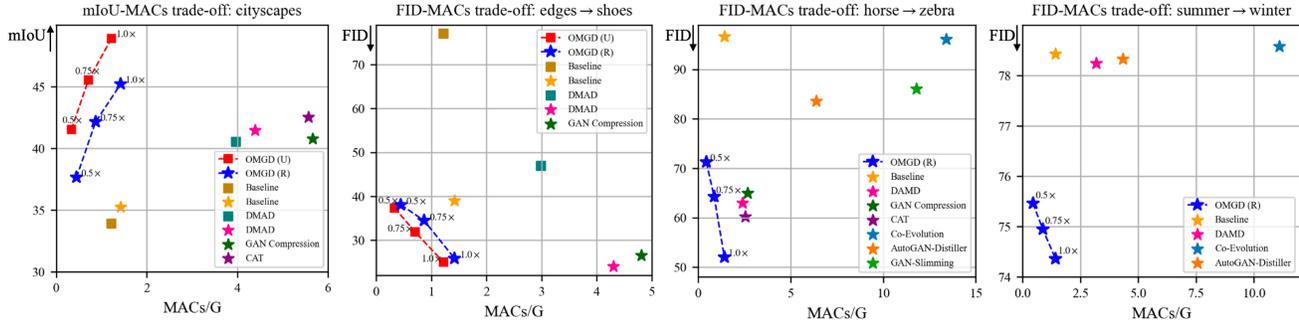
Figure 2: Performance-MACs trade-off between OMGD and existing competitive methods including GAN compression [30], CAT [24], DMAD [31], GAN-Slimming [49], AutoGAN-Distiller [13] and Co-Evolution [45]. □ denotes the U-Net style generator and ⋆ is the Res-Net style. OMGD significantly outperforms these methods with much less computational costs. "Baseline" denotes that the model is trained with naive GAN loss.

However, the above compression algorithms primarily exist threefold issues. Firstly, they tend to straightforwardly resort to the mature model compression techniques [7, 63, 19], which are not customized for GAN and lack exploration of complex characteristics and structures for GAN. Secondly, they usually formulate GAN compression as a multi-stages task. For example, [30] needs to pre-train, distillate, evolute, and fine-tune sequentially. The distillation-based methods [30, 31, 13, 9, 8, 49, 24] should pre-train a teacher generator and then distill the student one. An end-to-end approach is essential to reduce the complicated time and computational resources in the multi-stages setting. Thirdly, the current state-of-the-art methods are still burdened with high computational costs. For instance, the best model [31] requires 3G MACs, which is relatively high for deployment on lightweight edge devices.

To overcome the above issues, we craft to propose a novel *Online Multi-Granularity Distillation (OMGD)* framework for learning efficient GANs. We abandon the complex multi-stage compression process and design a GAN-oriented online distillation strategy to obtain the compressed model in one step. We can excavate potential complementary information from multiple levels and granularities to assist in the optimization of compressed models. These concepts can be regarded as auxiliary supervision cues, which is very critical to break through the bottleneck of capacity for models with low computational costs. The contributions of OMGD can be summarized as follows:

- To the best of our knowledge, we offer the first attempt to popularize distillation to an online scheme in the field of GAN compression and optimize the student generator in a discriminator-free and ground-truth-free setting. This scheme trains the teacher and student alternatively, promoting these two generators iteratively and progressively. The progressively optimized teacher generator helps to warm up the student and guide the optimization direction step by step.

- We further extend the online distillation strategy into a multi-granularity scheme from two perspectives. On the one hand, we employ different structure based teacher generators to capture more complementary cues and enhance visual fidelity from more diversified dimensions. On the other hand, besides the concepts of the output layer, we also transfer the channel-wise granularity information from intermediate layers to play as additional supervisory signals.

- Extensive experiments on widely-used datasets (i.e., horse→zebra [66], summer→winter [66], edges→shoes [60] and cityscapes [12]) demonstrate that OMGD can reduce the computation of two essential conditional GAN models including pix2pix [23] and CycleGAN [66] by **40×** regarding MACs, without loss of the visual fidelity of generated images. It reveal that OMGD is efficient and robust for various benchmark datasets, diverse conditional GAN, network architectures as well as problem settings (paired or unpaired). Compared with the existing competitive approaches, OMGD helps to obtain better image quality with much less computational costs (see Figure 1 and 2). Furthermore, OMGD 0.5× (only requires **0.333G MACs**) successes to achieve impressive results, which provides a feasible solution for deployment on resource-constrained devices and even breaks the barriers to real-time image translation on mobile devices.

2. Related Work

2.1. GANs and GAN Compression

Generative Adversarial Networks (GANs) [16] have obtained impressive results on a series of computer vision tasks, such as image-to-image translation [66, 23, 10, 26, 11, 47, 35], image generation [39, 40, 34, 64, 5, 4] and image inpainting [36, 58, 48, 62]. Specifically, Pix2Pix
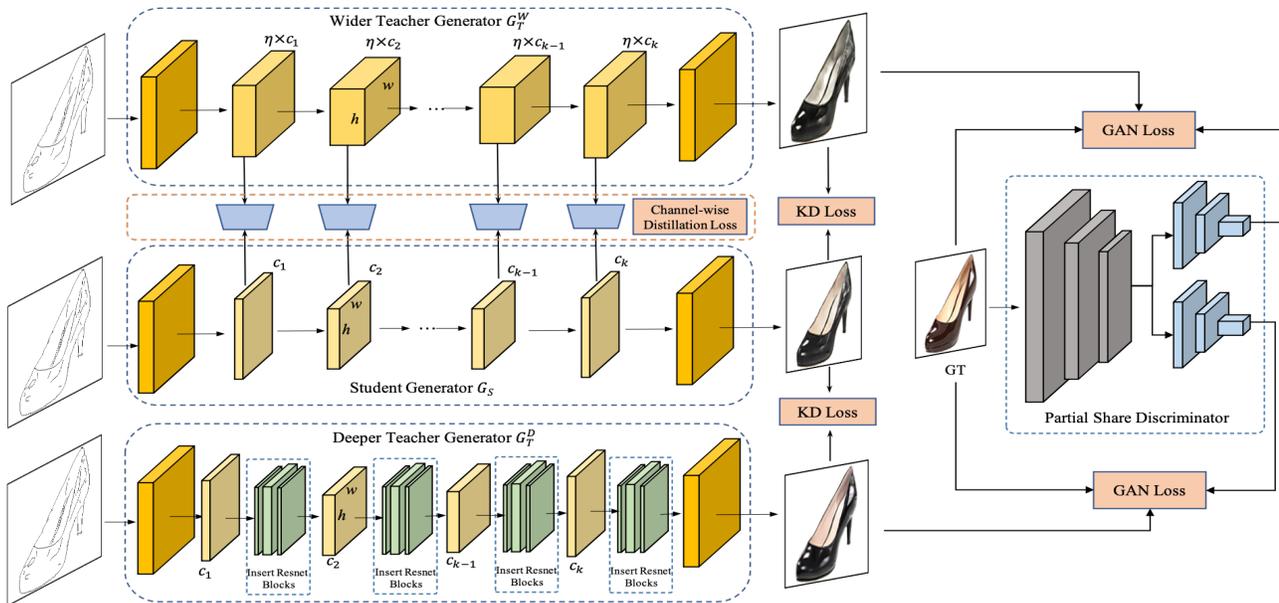
Figure 3: The pipeline of Online Multi-Granularity Distillation framework. The student generator G_S only leverages the complementary teacher generators (G_T^W and G_T^D) for optimization and can be trained in the discriminator-free and ground-truth-free setting. This framework transfers different levels concepts from the intermediate layers and output layer to perform the knowledge distillation. The whole optimization is conducted on a online distillation scheme. Namely, G_T^W, G_T^D and G_S are optimized simultaneously and progressively.

[23] conducted a min-max game between a generator and a discriminator to employ paired training data for image-to-image translation. CycleGAN [66] further extended the capacity of GANs for image translation in a weakly-supervised setting, where no paired data are leveraged in the training stage. Although various GAN methods have achieved impressive successes recently, they tend to occupy a growing number of memory and computational costs [10, 35, 5] to support their powerful performance, which conflicts the deployment on resource-constrained devices.

Recently, GAN-oriented compression has become an essential task due to its potential applications in the field of lightweight device deployment. Shu *et al*. [45] presented the first preliminary study on introducing the co-evolutionary algorithm for removing redundant filters to compress CycleGAN. Fu *et al*. [13] employed AutoML approaches and searched an efficient generator with the guide of the target computational resource constraints. Wang *et al*. [49] proposed a unified GAN compression optimization framework including model distillation, channel pruning and quantization. Li *et al*. [30] designed a "once-for-all" generator which decouples the model training and architecture search via weight sharing. Li *et al*. [31] proposed a differentiable mask and a co-attention distillion algorithm to learn effective GAN. Jin *et al*. [24] proposed a one-step pruning algorithm to search a student model from the teacher model. In this work, we design an Online Multi-

Granularity Distillation (OMGD) scheme. By introducing multi-granularity knowledge guidance, the student generator can be enhanced by leveraging the complementary concepts from diverse teachers and layers, which intrinsically improves the capacity of the compressed model.

2.2. Knowledge Distillation

Knowledge Distillation (KD) [19] is a fundamental compression technique, where a smaller student model is optimized under the effective information transfer and supervision of a larger teacher model or ensembles [6]. Hinton [19] performed knowledge distillation via minimizing the distance between the output distribution statistics between student and teacher network. In this way, the student network attempts to learn dark knowledge [19] that contains the similarities between different classes, which can not be provided by the ground truth labels. Romero *et al*. [41] further took advantage of the concepts of feature maps from the intermediate layers to enhance the performance of the student network. Zhou *et al*. [65] presented that each channel of the feature map corresponds to a visual pattern, so they focus on transferring attention concepts [53, 54, 55] of feature map from each channel in intermediate layers.

Moreover, You *et al*. [59] revealed that multiple teacher networks can provide more comprehensive knowledge for learning a more effective student network. MEAL [44] compressed large and complex trained ensembles into a sin-

gle network, which employs an adversarial-based learning strategy to guide the pre-defined student network to transfer knowledge from teacher models. Offline knowledge distillation requires a pre-trained teacher model in the stage of optimizing, while online KD simultaneously optimizes the teacher and student network or just a group of student peers [51]. Anil *et al.* [2] trained two networks with the identical architecture parallelly and these two networks play the role of student and teacher iteratively. In this paper, we employ the multi-granularity based online distillation scheme, which aims to learn an effective student model from complementary structure of the teacher generators and the knowledge from diverse layers.

3. Methodology

In this section, we first introduce the proposed online GAN distillation framework, where the student generator is not bound to the discriminator and attempts to learn concepts directly from the teacher models. The multi-granularity distillation scheme is presented in section 3.2 The multi-granularity concepts [32] are captured via complementary teacher generators and knowledge from diverse layers. We illustrate the whole pipeline of the OMGD framework in Figure 3.

3.1. Online GAN Distillation

Recently, a series of distillation based GAN compression [30, 31, 13, 9, 8, 49] employ the offline distillation scheme that leverages a pre-trained teacher generator to optimize the student generator. In this paper, we propose a GAN-oriented online distillation algorithm to address three critical issues in the offline distillation. Firstly, the student generator in the conventional offline distillation method should maintain a certain capacity to keep the dynamic balance with the discriminator to avoid model collapse [37, 43] and vanishing gradients [3]. However, our student generator is no longer deeply bound with the discriminator, which can train more flexibly and obtain further compression. Secondly, the pre-trained teacher generator fail to guide the student on how to learn information progressively and is easy to cause over-fitting in the training stage [17, 27]. Nevertheless, our teacher generator helps to warm up student generator and guide the direction of optimization step by step. Thirdly, it is not effortless to select a suitable pre-trained teacher generator due to the evaluation metrics are subjective [43]. However, our method does not require a pre-trained model and this selection problem is solved.

Teacher Generator. We follow the loss functions and training setting in [23, 66] to train the teacher generator G_T and discriminator D. G_T aims to learn a function to map data from the source domain X to a target domain Y. We take Pix2Pix [23] as example, it uses paired data($\{x_i, y_i\}_{i=1}^N$, where $x_i \in X$ and $y_i \in Y$) to optimize

networks. The generator G_T is trained to map x_i to y_i while the discriminator D is trained to distinguish the fake images generated by G_T from the real images. The objective is formalized as:

$$\mathcal{L}_{GAN}(G_T, D) = \mathbb{E}_{x,y}[\log D(x, y)] \\ + \mathbb{E}_x[\log(1 - D(x, G_T(x)))]. \quad (1)$$

Moreover, a reconstruction loss is introduced to push the output of G_T output to be close to the ground truth y:

$$\mathcal{L}_{Recon}(G_T) = \mathbb{E}_{x,y}[\| y - G_T(x) \|_1]. \quad (2)$$

The whole objective in the GAN setting is defined as:

$$G_T^* = \arg \min_{G_T} \max_D \mathcal{L}_{GAN}(G_T, D) + \mathcal{L}_{Recon}(G_T). \quad (3)$$

Student Generator. In the proposed GAN-oriented online distillation scheme, the student generator G_S only leverages the teacher network G_T for optimization and can be trained in the discriminator-free setting. The optimization of G_S does not require the ground-truth labels y simultaneously. Namely, G_S merely learns the output of the larger capacity generator with a similar structure (G_T), which greatly reduces the difficulty of fitting y directly. Specifically, we back-propagate the distillation loss between G_T and G_S in every iterative step. In this way, G_S can mimic the training process of G_T to learn progressively.

Denote the output of G_T/G_S as p_t/p_s, we use *Structural Similarity (SSIM) Loss* [52] and *Perceptual Losses* [25] to measure the difference between p_t and p_s. SSIM Loss [52] is sensitive to local structural changes, which is similar to human visual system (HVS). Given p_s, p_t, SSIM Loss calculates the similarity of two images by:

$$\mathcal{L}_{SSIM}(p_t, p_s) = \frac{(2\mu_t \mu_s + C_1)(2\sigma_{ts} + C_2)}{(\mu_t^2 \mu_s^2 + C_1)(\sigma_t^2 + \sigma_s^2 + C_2)}, \quad (4)$$

where μ_s, μ_t are mean values for luminance estimation, σ_s^2, σ_t^2 are standard deviations for contrast, σ_{ts} is covariance for the structural similarity estimation. C_1, C_2 are constants to avoid zero denominator.

Perceptual loss [25] consists of feature reconstruction loss $\mathcal{L}_{feature}$ and style reconstruction loss $\mathcal{L}_{style}$. $\mathcal{L}_{feature}$ encourages p_t and p_s to have similar feature representations, which are measured by a pre-trained VGG network [46] ϕ. $\mathcal{L}_{feature}$ is formalized as:

$$\mathcal{L}_{feature}(p_t, p_s) = \frac{1}{C_j H_j W_j} \| \phi_j(p_t) - \phi_j(p_s) \|_1, \quad (5)$$

where $\phi_j(x)$ is the activation of the j-th layer of ϕ for the input x. $C_j \times H_j \times W_j$ is the dimensions of $\phi_j(x)$.

$\mathcal{L}_{style}$ is introduced to penalize the differences in style characteristic, such as color, textures, common pattern, and so on [14]. The $\mathcal{L}_{style}$ can be calculated as:

$$\mathcal{L}_{style}(p_t, p_s) = \| G_j^\phi(p_t) - G_j^\phi(p_s) \|_1, \quad (6)$$

where $G_j^\phi(x)$ is the *Gram matrix* of the j-th layer activation in the VGG network.

Furthmore, the total variation loss $\mathcal{L}_{TV}$ [42] is introduced to encourages spatial smoothness in the generated image. We use four hyperparameters $\lambda_{SSIM}, \lambda_{feature}, \lambda_{style}, \lambda_{TV}$ to achieve the balance between the above losses, so the total online KD loss $\mathcal{L}_{KD}(p_t, p_s)$ is computed by:

$$\begin{aligned}
\mathcal{L}_{KD}(p_t, p_s) =& \lambda_{SSIM}\mathcal{L}_{SSIM} + \lambda_{feature}\mathcal{L}_{feature} \\
& + \lambda_{style}\mathcal{L}_{style} + \lambda_{TV}\mathcal{L}_{TV}
\end{aligned} \quad (7)$$

3.2. Multi Granularity Distillation Scheme

Based on the novel online GAN distillation technique, we further extend our method into a multi-granularity scheme from two perspectives: the complementary structure of the teacher generator, and the knowledge from diverse layers. The whole pipeline of the online multi-granularity distillation (OMGD) framework is depicted in Figure 3, we use a wider teacher generator G_T^W and a deeper teacher generator G_T^D to formalize a multi-objective optimization task for G_S. In addition to the output layer of the teacher generators, we also mine knowledge concepts from the intermediate layers via channel distillation loss [65].

Multiple Teachers Distillation. A different structure based teacher generator helps to capture more complementary image cues from the ground truth labels and enhance image translation performance from a different perspective [59]. Moreover, the multiple teachers distillation setting can further relieve the issue of over-fitting. We expand the student model into the teacher model from two complementary dimensions, i.e., depth and width. Given a student generator G_S, we expand the channel of G_S to obtain a wider teacher generator G_T^W. Specifically, each channel of the convolution layers is multiplied by an channel expand factor η. On the other hand, we insert several resnet blocks after every downsample and upsample layers into G_S to build a deeper teacher generator G_T^D, which has a comparable capacity with G_T^W.

As is illustrated in Figure 3, a partial share discriminator is designed to share the first several layers and separate two branches to get the discriminator output for G_T^W and G_T^D, respectively. This shared design not only offers high flexibility of discriminators but also leverages the similar characteristic of the input image to improve the training of generators [20]. We directly combine two distillation losses provides by the complementary teacher generators as the KD loss in multiple teachers setting:

$$\begin{aligned}
\mathcal{L}_{KD_{multi}}&(p_t^w, p_t^d, p_s) \\
&= \mathcal{L}_{KD}(p_t^w, p_s) + \mathcal{L}_{KD}(p_t^d, p_s),
\end{aligned} \quad (8)$$

where p_t^w and p_t^d are the activation of the output layer of G_T^W and G_T^D, respectively.

Intermediate Layers Distillation. The concepts of the output layer fail to take into account of more intermediate details of the teacher network, so we further transfer the channel-wise granularity information as an additional supervisory signal to promote G_S. Specifically, we compute the channel-wise attention weight [22, 65] to measure the importance of each channel in a feature map. The attention weight w_c is defined as:

$$w_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j), \quad (9)$$

where u_c denotes the cth channel of the feature map. Then a 1×1 convolution layer is concatenated to the intermediate layers of G_S to expand the number of channel and the channel distillation (CD) loss is calculated as:

$$\mathcal{L}_{CD}(G_T^W, G_S) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{c} (w_{t_w}^{ij} - w_s^{ij})^2}{c} \right), \quad (10)$$

where n is the number of feature maps be sampled, c is the channel number of the feature maps. w^{ij} is the attention weight of j-th channel of i-th feature map.

To sum up, the whole online multi-granularity distillation objective is formalized as:

$$\begin{aligned}
\mathcal{L}(G_T^W&, G_T^D, G_S) \\
&= \lambda_{CD}\mathcal{L}_{CD}(G_T^W, G_S) + \mathcal{L}_{KD_{multi}}(p_t^w, p_t^d, p_s)
\end{aligned} \quad (11)$$

4. Experiments

4.1. Experimental Settings

We follow the models, datasets and evaluation metrics used in previous works [13, 30, 31, 45] for a fair comparison.

Models. We conduct the experiments on Pix2Pix [23] and CycleGAN [66]. Specifically, we adopt the original U-Net style generator [23] and the Res-Net style generator in [30] for Pix2Pix [23] model. The Res-Net style generator employs depthwise convolution and pointwise convolution [21] to achieve a better performance-computation trade-off. We only use the Res-Net style generator [30] to conduct the experiments on CycleGAN model.

Datasets and Evaluation Metrics. We evaluate Pix2Pix model on edges→shoes [60] and cityscapes [12] dataset. CycleGAN model is measured on horse→zebra [66] and summer→winter [66]. On cityscapes, we use the DRN-D-105 [61] to segment the generated images and calculate the mIoU (mean Intersection over Union) as evaluation metric. Higher mIoU implies the generated images are more realistic. We adopt FID (Frechet Inception Distance) [18] to evaluate the image on other datasets and smaller FID means the generation performance is more convincing.

Table 1: The performance comparison with state-of-the-art methods in Pix2Pix model.

Dataset	Generator Style	Method	MACs	#Parameters	FID (↓)
edges→shoes	Res-Net	Original [23]	56.80G (1.0×)	11.30M (1.0×)	24.18
		GAN-Compression [30]	4.81G (11.8×)	0.70M (16.3×)	26.60
		DMAD [31]	4.30G (13.2×)	0.54M (20.9×)	24.08
		OMGD 1.0×	1.408G (40.3×)	0.137M (82.5×)	25.88
		OMGD 1.5×	**2.904G (19.6×)**	**0.296M (38.2×)**	**21.41**
	U-Net	Original [23]	18.60G (1.0×)	54.40M (1.0×)	34.31
		DMAD [31]	2.99G (6.2×)	2.13M (25.5×)	46.95
		OMGD 0.5×	0.333G (55.9×)	0.852M (63.8×)	37.34
		OMGD 0.75×	0.707G (26.3×)	1.916M (28.4×)	32.19
		OMGD 1.0×	**1.219G (15.3×)**	**3.404M (16.0×)**	**25.00**

Dataset	Generator Style	Method	MACs	#Parameters	mIoU (↑)
cityscapes	Res-Net	Original [23]	56.80G (1.0×)	11.30M (1.0×)	44.32
		GAN-Compression [30]	5.66G (10×)	0.71M (15.9×)	40.77
		DMAD [31]	4.39G (12.9×)	0.55M (20.5×)	41.47
		CAT [24]	5.57G (10.2×)	-	42.53
		OMGD 1.0×	1.408G (40.3×)	0.137M (82.5×)	45.21
		OMGD 1.5×	**2.904G (19.6×)**	**0.296M (38.2×)**	**45.89**
	U-Net	Original [23]	18.60G (1.0×)	54.40M (1.0×)	42.71
		DMAD [31]	3.96G (4.7×)	1.73M (31.4×)	40.53
		OMGD 0.5×	0.333G (55.9×)	0.852M (63.8×)	41.54
		OMGD 0.75×	0.707G (26.3×)	1.916M (28.4×)	45.52
		OMGD 1.0×	**1.219G (15.3×)**	**3.404M (16.0×)**	**48.91**

Implementation datails. The channel expand factor η is set to 4 in this paper. We set the learning rate as 0.0002 in the beginning and decay to zero linearly in the experiments. For Res-Net style generator, batch size is set to 4 on edges→shoes and 1 on other dataset. Batch size is fixed to 4 in all expreiments for U-Net generator. The update interval numbers n on edges→shoes, cityscapes, horse→zebra and summer→winter are 1, 3, 4, 4 respectively.

For CycleGAN, we evaluate the teacher generator every m epochs and update the performance-best generator as G_T to optimize G_S. In this way, we avoid notorious instability of training CycleGAN and encourage G_S to learn from the best teacher model. m is set to 10 and 6 for horse→zebra and summer→winter, respectively.

4.2. Experimental Results

4.2.1 Comparison with state-of-the-art methods

In this section, we compare OMGD with several state-of-the-art methods in terms of computation cost, model size and generation quality. We compare the performance of Pix2Pix and CycleGAN respectively.

Pix2Pix. The experimental results of Pix2Pix model are shown in Table 1, which can summarized as the following observerations: 1) OMGD is robust for both style generators and significantly outperforms the state-of-the-art methods with much less computational costs. 2) OMGD with

Res-Net style generator (dubbed as, OMGD(R)) achieves comparable performance to the original model when the MACs are compressed by **40.3×** and the parameters are compressed by **82.5×**. Compared with the current best method, i.e., CAT, OMGD(R) 1.0× boosts the mIoU from 42.53 to 45.21 (6.3% improvement) with only a quarter of the computational costs on cityscapes. Furthmore, although OMGD(R) 1.5× is compressed **19.6×** MACs and **38.2×** memory, it successes to establishes the state-of-the-art performance. 3) It is arduous to compress U-Net style generator due to its U-shape architecture and concatenate operation. OMGD with U-Net style generator (dubbed as, OMGD(U)) compresses the original model by **15.3×** and reduces the FID by 9.31 on edges→shoes. With less than half of MACs of DMAD [31], OMGD(U) 1.0× decreases the FID from 46.95 to 25.0 on edges→shoes and obtains 19.3% improvement in terms of mIoU on cityscapes. Moreover, OMGD(U) 0.5× and 0.75× also achieve impressive results, and OMGD(U) 0.75× can obatin the state-of-the-art compression performance with merely **0.707G** MACs.

CycleGAN. We follow previous works [30, 13, 24, 15, 49] to use the Res-Net style generator to conduct the experiments on CycleGAN, and the results are shown in Table 2. On the one hand, OMGD(R) outperforms the original model by a large margin although with **40.3×** MACs compression and **82.5×** parameters compression. For ex-

Table 2: The performance comparison with state-of-the-art methods in CycleGAN model.

Dataset	Method	MACs	#Parameters	FID (↓)
horse→zebra	Original [66]	56.80G (1.0×)	11.30M (1.0×)	61.53
	Co-Evolution [45]	13.40G (4.2×)	-	96.15
	GAN-Slimming [49]	11.25G (23.6×)	-	86.09
	Auto-GAN-Distiller [13]	6.39G (8.9×)	-	83.60
	GAN-Compression [30]	2.67G (21.3×)	0.34M (33.2×)	64.95
	DMAD [31]	2.41G (23.6×)	0.28M (40.0×)	62.96
	CAT [24]	2.55G (22.3×)	-	60.18
	OMGD (Ours)	**1.408G (40.3×)**	**0.137M (82.5×)**	**51.92**
summer→winter	Original [66]	56.80G (1.0×)	11.30M (1.0×)	79.12
	Co-Evolution [45]	11.10G (5.1×)	-	78.58
	Auto-GAN-Distiller [13]	4.34G (13.1×)	-	78.33
	DMAD [31]	3.18G (17.9×)	0.30M (37.7×)	78.24
	OMGD (Ours)	**1.408G (40.3×)**	**0.137M (82.5×)**	**73.79**

Table 3: Ablation Study on Pix2Pix model.

Dataset	Method	FID (↓)
edges→shoes	Baseline	77.07
	Ours w/o OD	26.19
	Ours w/o DT	33.88
	Ours w/o CD	26.62
	Ours	**25.00**

Dataset	Method	mIoU (↑)
cityscapes	Baseline	33.90
	Ours w/o OD	45.76
	Ours w/o DT	44.04
	Ours w/o CD	48.12
	Ours	**48.91**

Table 4: Ablation Study on CycleGAN model.

Dataset	Method	FID (↓)
horse→zebra	Baseline	96.72
	Ours w/o OD	77.09
	Ours w/o CD	61.21
	Ours	**51.92**
summer→winter	Baseline	78.43
	Ours w/o OD	76.48
	Ours w/o CD	75.47
	Ours	**73.79**

ample, OMGD(R) reduces FID from 61.53 to 51.92 on horse→zebra and 79.12 to 73.79 on summer→winter. On the other hand, OMGD(R) significantly surpasses the competitive methods in terms of performance (FID) or computational costs (MACs), and establishes new state-of-the-art performance on both datasets.

4.2.2 Ablation Study

We directly train the student generator via the conventional GAN loss and report its results as the "Baseline" in Table 3 and Table 4. As can be observed, our method surpasses "Baseline" by a large margin. For example, it declines FID from 77.07 to 25.00 on edges→shoes and increases mIoU from 33.90 to 48.91 on cityscapes. To further demonstrate the effectiveness of several essential components in OMGD, we perform extensive ablation studies. The experiments of ablation study are conducted on U-Net style generator for Pix2Pix and Res-Net style generator for CycleGAN,

Analysis of online distillation stage. To evaluate the significance of the online distillation scheme, we design a variant (abbreviated as "Ours w/o OD") to optimize the model with the offline two-stage distillation setting. As shown in Table 3 and 4, removing the online distillation stage leads to an noticeable drop in performance. For example, "Ours w/o OD" declines mIoU to 45.76 on cityscapes, with a decrease of 6.4% when compared with our approach. It indicates that the online training scheme helps to guide the optimization process to achieve more impressive results.

Analysis of complementary teachers setting. To investigate the effectiveness of complementary teachers setting, we design a variant "Ours w/o DT" that removes the deeper teacher generator and only employs a wider one for optimization. As summarized in Table 3, our method attempts to obtain more promising results compared with "Ours w/o DT" on both benchmarks. It indicates that the complementary teacher setting significantly improves the capacity of the student generator. It is worth notice that the unstable training process of CycleGAN causes confusion for the deeper teacher generator G_T^D, hence we only leverage the wider teacher generator on CycleGAN.

Analysis of multiple distillation layers. To delve deep into the significance of multiple distillation layers, we de-
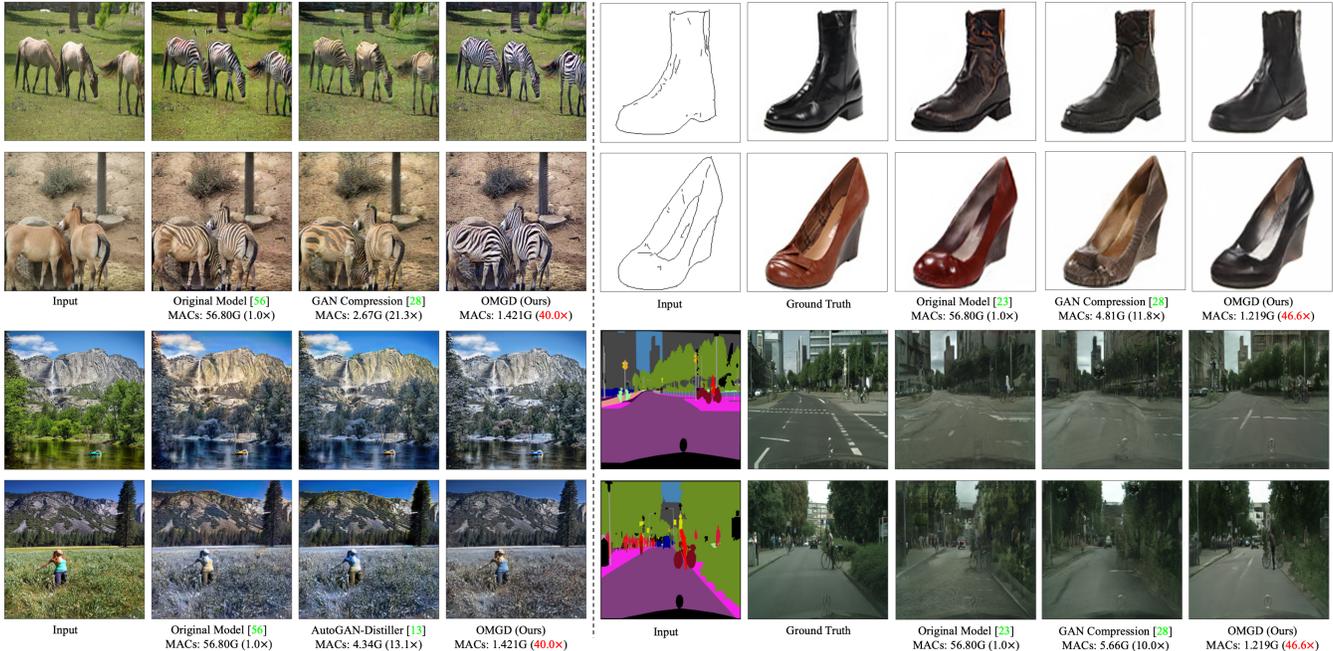
Figure 4: Qualitative compression results on horse→zebra, summer→winter, cityscapes and edges→shoes. OMGD preserves the fidelity while significantly reducing the computation.

Table 5: Latency Speedup on Mobile Phones (CPU).

Device	original	OMGD(U) 1.0×	
	Latency	Latency	MAC
HuaWei P20	416.73ms	43.00ms (9.7×)	15.3 ×
Mi 10	140.80ms	14.01ms (10.0 ×)	15.3 ×

sign a variant (denote as "Ours w/o CD") to remove the channel-wise distillation. As shown in Table 3 and Table 4, "Ours w/o CD" gets a less prominent performance, which indicates that concepts from intermediate layers can be viewed as the auxiliary supervision to assist training. By introducing multiple distillation layers for distillation, our method manages to obtain 6.5%, 1.6%, 15.1% and 1.5% performance improvement on four datasets, respectively.

4.2.3 Latency Speedup

We report the CPU latency results on two mobile phones (i.e., Huawei P20 and Mi 10) using tflite toolkits. As is shown in Table 5, our framework helps to obtain significant acceleration in the inference procedure. For example, OMGD(U) 1.0× contributes to reducing latency from 140.8ms to 14.01ms, with a **90%** latency decline. It demonstrates that our framework provides a solution for real-time image translation.

4.2.4 Qualitative Results

We depict the visualization results of OMGD and the state-of-the-art methods in Figure 4, which demonstrates the ef-

fectiveness of OMGD. As shown, our method helps to obtain 40.3-46.6 × MACs reductions with nearly no visual fidelity loss. For example, Our $40\times$ compressed model can generate natural zebra stripes on horse→zebra dataset, while [30] and original model still retain the color of the input horse. OMGD attempts to transfer the background style smoothly, while preserves the essential elements in the foreground on summer→winter. For Pix2Pix, OMGD contributes to capturing the textural details of the cloth fabric and the shine of the leather fabric on edges→shoes. Furthermore, OMGD shows superiority in the processing of pavement features, such as roughness and lane line.

5. Conclusion

In this paper, we propose an online multi-granularity distillation (OMGD) technique for learning lightweight GAN. The GAN-oriented online scheme is introduced to alternately promote the teacher and student generator, and the teacher helps to warm up the student and guide the optimization direction step by step. OMGD further takes good advantage of multi-granularity concepts from complementary teacher generators and auxiliary supervision signals from different layers. Extensive experiments demonstrate that OMGD attempts to compress Pix2Pix and CycleGAN into extremely low computational costs without obvious visual fidelity loss, which provides a feasible solution for GAN deployment on resource-constrained devices.

References

[1] A. Aguinaldo, Ping-Yeh Chiang, Alexander Gain, Ameya Patil, Kolten Pearson, and S. Feizi. Compressing gans using knowledge distillation. *ArXiv*, abs/1902.00159, 2019.

[2] R. Anil, G. Pereyra, Alexandre Passos, Róbert Ormándi, G. Dahl, and Geoffrey E. Hinton. Large scale distributed neural network training through online distillation. *ArXiv*, abs/1804.03235, 2018.

[3] Martín Arjovsky, Soumith Chintala, and L. Bottou. Wasserstein gan. *ArXiv*, abs/1701.07875, 2017.

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

[5] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. *ArXiv*, abs/1809.11096, 2019.

[6] Cristian Bucila, R. Caruana, and Alexandru Niculescu-Mizil. Model compression. In *KDD '06*, 2006.

[7] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. In *International Conference on Learning Representations*, 2020.

[8] Ting-Yun Chang and Chi-Jen Lu. Tinygan: Distilling biggan for conditional image generation. *ArXiv*, abs/2009.13829, 2020.

[9] 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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3585–3592, Apr. 2020.

[10] Yang Chen, Yu-Kun Lai, and Yong-Jin Liu. Cartoongan: Generative adversarial networks for photo cartoonization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[11] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, M. Enzweiler, Rodrigo Benenson, Uwe Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[13] Y. Fu, Wuyang Chen, Haotao Wang, Haoran Li, Yingyan Lin, and Zhangyang Wang. Autogan-distiller: Searching to compress generative adversarial networks. In *ICML*, 2020.

[14] Leon A. Gatys, Alexander S. Ecker, and M. Bethge. A neural algorithm of artistic style. *ArXiv*, abs/1508.06576, 2015.

[15] X. Gong, S. Chang, Yifan Jiang, and Zhangyang Wang. Autogan: Neural architecture search for generative adversarial networks. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3223–3233, 2019.

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[17] Qiushan Guo, Xinjiang Wang, Yichao Wu, Zhipeng Yu, Ding Liang, Xiaolin Hu, and Ping Luo. Online knowledge distillation via collaborative learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[20] L. Hou, Ze-Huan Yuan, Lei Huang, Hua-Wei Shen, Xueqi Cheng, and Changhu Wang. Slimmable generative adversarial networks. *ArXiv*, abs/2012.05660, 2020.

[21] A. Howard, Menglong Zhu, Bo Chen, D. Kalenichenko, W. Wang, Tobias Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv*, abs/1704.04861, 2017.

[22] Jie Hu, L. Shen, Samuel Albanie, G. Sun, and Enhua Wu. Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:2011–2023, 2020.

[23] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017.

[24] Qing Jin, Jian Ren, Oliver J. Woodford, and Jiazhuo Wang. Teachers do more than teach: Compressing image-to-image models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[25] J. Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[27] X. Lan, Xiatian Zhu, and S. Gong. Knowledge distillation by on-the-fly native ensemble. In *NeurIPS*, 2018.

[28] Huixia Li, Chenqian Yan, Shaohui Lin, Xiawu Zheng, Baochang Zhang, Fan Yang, and Rongrong Ji. Pams: Quantized super-resolution via parameterized max scale. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 564–580. Springer, 2020.

[29] Jiashi Li, Qi Qi, Jingyu Wang, Ce Ge, Yujian Li, Zhangzhang Yue, and Haifeng Sun. Oicsr: Out-in-channel sparsity regularization for compact deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7046–7055, 2019.

[30] Muyang Li, J. Lin, Yaoyao Ding, Zhijian Liu, Jun-Yan Zhu, and Song Han. Gan compression: Efficient architectures for

interactive conditional gans. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5283–5293, 2020.

[31] Shaojie Li, Mingbao Lin, Yan Wang, M. Xu, Feiyue Huang, Yongjian Wu, Ling Shao, and Rongrong Ji. Learning efficient gans using differentiable masks and co-attention distillation. *ArXiv*, abs/2011.08382, 2020.

[32] Yingying Li, Jie Wu, Xue Bai, Xipeng Yang, Xiao Tan, Guanbin Li, Shilei Wen, Hongwu Zhang, and Errui Ding. Multi-granularity tracking with modularlized components for unsupervised vehicles anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 586–587, 2020.

[33] Mingbao Lin, Rongrong Ji, Shaojie Li, Yan Wang, Yongjian Wu, Feiyue Huang, and Qixiang Ye. Network pruning using adaptive exemplar filters. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[34] Takeru Miyato, T. Kataoka, Masanori Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *ArXiv*, abs/1802.05957, 2018.

[35] 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 (CVPR)*, June 2019.

[36] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[37] Xue Bin Peng, Angjoo Kanazawa, Sam Toyer, Pieter Abbeel, and Sergey Levine. Variational discriminator bottleneck: Improving imitation learning, inverse RL, and GANs by constraining information flow. In *International Conference on Learning Representations*, 2019.

[38] Qi Qi, Yan Lu, Jiashi Li, Jingyu Wang, Haifeng Sun, and Jianxin Liao. Learning low resource consumption cnn through pruning and quantization. *IEEE Transactions on Emerging Topics in Computing*, 2021.

[39] A. Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2016.

[40] Scott E. Reed, Zeynep Akata, Xinchen Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016.

[41] A. Romero, Nicolas Ballas, S. Kahou, Antoine Chassang, C. Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *CoRR*, abs/1412.6550, 2015.

[42] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992.

[43] Tim Salimans, Ian J. Goodfellow, W. Zaremba, Vicki Cheung, A. Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, 2016.

[44] Zhiqiang Shen, Z. He, and X. Xue. Meal: Multi-model ensemble via adversarial learning. In *AAAI*, 2019.

[45] H. Shu, Yunhe Wang, Xu Jia, Kai Han, H. Chen, Chunjing Xu, Q. Tian, and Chang Xu. Co-evolutionary compression for unpaired image translation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3234–3243, 2019.

[46] K. Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

[47] Hao Tang, Dan Xu, Nicu Sebe, Yanzhi Wang, Jason J. Corso, and Yan Yan. Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[48] Ziyu Wan, B. Zhang, Dongdong Chen, P. Zhang, D. Chen, Jing Liao, and Fang Wen. Bringing old photos back to life. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2744–2754, 2020.

[49] Haotao Wang, Shupeng Gui, Haichuan Yang, J. Liu, and Zhangyang Wang. Gan slimming: All-in-one gan compression by a unified optimization framework. *ArXiv*, abs/2008.11062, 2020.

[50] Kuan Wang, Zhijian Liu, Yujun Lin, Ji Lin, and Song Han. Haq: Hardware-aware automated quantization with mixed precision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[51] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2021.

[52] Zhou Wang, A. Bovik, H. R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:600–612, 2004.

[53] Jie Wu, Haifeng Hu, and Yi Wu. Image captioning via semantic guidance attention and consensus selection strategy. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4):1–19, 2018.

[54] Jie Wu, Haifeng Hu, and Liang Yang. Pseudo-3d attention transfer network with content-aware strategy for image captioning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 15(3):1–19, 2019.

[55] Jie Wu, Siya Xie, Xinbao Shi, and Yaowen Chen. Global-local feature attention network with reranking strategy for image caption generation. In *CCF Chinese Conference on Computer Vision*, pages 157–167. Springer, 2017.

[56] Tasweer Ahmad Xuefeng Xiao, Yafeng Yang and Tianhai Chang Lianwen Jin. Design of a very compact cnn classifier for online handwritten chinese character recognition using dropweight and global pooling. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 891–895. IEEE, 2017.

[57] Yafeng Yang Weixin Yang Jun Sun Tianhai Chang Xuefeng Xiao, Lianwen Jin. Building fast and compact convolutional neural networks for offline handwritten chinese character recognition. *Pattern Recognition*, 72:72–81, 2017.

[58] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[59] Shan You, C. Xu, and D. Tao. Learning from multiple teacher networks. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.

[60] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 192–199, 2014.

[61] F. Yu, V. Koltun, and T. Funkhouser. Dilated residual networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 636–644, 2017.

[62] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018.

[63] J. Yu, L. Yang, N. Xu, Jianchao Yang, and T. Huang. Slimmable neural networks. *ArXiv*, abs/1812.08928, 2019.

[64] Han Zhang, Ian J. Goodfellow, Dimitris N. Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *ICML*, 2019.

[65] Zaida Zhou, Chaoran Zhuge, Xinwei Guan, and Wen Liu. Channel distillation: Channel-wise attention for knowledge distillation. *ArXiv*, abs/2006.01683, 2020.

[66] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.