Information-Theoretic GAN Compression with Variational Energy-based Model

Minsoo Kang[1] **Hyewon Yoo**[2] **Eunhee Kang**[3] **Sehwan Ki**[3] **Hyong-Euk Lee**[3] **Bohyung Han**[1,2]

[1]ECE & [2]IPAI, Seoul National University
[3]Samsung Advanced Institute of Technology (SAIT)
{kminsoo, yoohyewony, bhhan}@snu.ac.kr
{eunhee.kang, sh1004.ki, hyongeuk.lee}@samsung.com

A Appendix

This appendix first discusses the related works to cooperative learning methods. Second, we describe the objective function for knowledge distillation employed in GAN compression algorithms [1–5]. Third, we describe the implementation details of VEM in terms of the architecture design of the energy-based models. Finally, we present the additional qualitative results of VEM compared with the state-of-the-art methods.

A.1 Related Work to Cooperative Learning Methods

The proposed method is related to cooperative learning mechanisms [6–9] in the sense that an energy-based model is employed for a student such as variational auto-encoder frameworks [8], flow-based models [9] or generic generators [6, 7] to achieve better generation performance. However, their usages and roles are fairly different since we optimize the energy-based model to minimize the KL divergence of the true conditional distribution from the variational one in order to precisely estimate mutual information for effective knowledge distillation. On the other hand, the cooperative learning methods simply learn the energy-based model to maximize the data likelihood. Moreover, our key contributions lie in 1) the information-theoretic problem formulation of GAN compression using the mutual information and 2) the introduction of EBMs to variational distributions and its successful application to a practical problem.

A.2 GAN Compression Baseline Algorithms

To validate the benefit and generality of VEM, we optimize student generators with algorithm-specific objective functions, $\mathcal{L}_{\text{algo}}$, by jointly considering the mutual information between teacher and student models for effective knowledge distillation, as shown in (10). We present the algorithm-specific loss functions for two online distillation approaches, OMGD [4] and GCC [5], and three offline methods, GAN-Compression [1], CAT [2], and CAGC [3].

A.2.1 OMGD [4]

The algorithm-specific loss, denoted by $\mathcal{L}_{\text{algo}}$, is defined for OMGD as follows:

$$\mathcal{L}_{\text{algo}} = \mathcal{L}_{\text{OMGD-KD}} + \lambda_{\text{CD}}\mathcal{L}_{\text{CD}} + \lambda_{\text{TV}}\mathcal{L}_{\text{TV}}, \tag{15}$$

where $\mathcal{L}_{\text{OMGD-KD}}$ and $\mathcal{L}_{\text{CD}}$ are the distillation losses for the final and intermediate outputs, respectively while $\mathcal{L}_{\text{TV}}$ denotes the total variation. The hyperparameters λ_{CD} and λ_{TV} determine the weights of the corresponding loss terms. OMGD [4] employs the channel distillation loss [10, 11] for the intermediate feature maps, which encourages the student network to mimic the channel-wise attention

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

maps in the intermediate layers of the teacher. Specifically, the channel distillation loss, $\mathcal{L}_{\text{CD}}$, is given by

$$\mathcal{L}_{\text{CD}} = \sum_{\ell=1}^{L} \mathbb{E}_{\boldsymbol{x}} \Big[\mathcal{L}_{\text{atten}} \big(G_\ell^t(\boldsymbol{x}; \phi^t), f_\ell(G_\ell^s(\boldsymbol{x}; \phi^s)) \big) \Big], \tag{16}$$

where $G_\ell^t(\cdot; \phi^t)$ and $G_\ell^s(\cdot; \phi^s)$ are the intermediate feature maps of the ℓ^{th} layer in the teacher and student networks, and $f_\ell(\cdot)$ denotes a 1×1 convolution operator to match the dimensionality of the two feature maps. In the above equation, the attention loss, $\mathcal{L}_{\text{atten}}(\cdot, \cdot)$, is given by

$$\mathcal{L}_{\text{atten}}(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{C} \| \text{GAP}(\boldsymbol{p}) - \text{GAP}(\boldsymbol{q}) \|_2^2, \tag{17}$$

where $\text{GAP}(\cdot)$ is an average pooling function over the spatial dimension and C is the number of channels in $\boldsymbol{p}$ and $\boldsymbol{q}$, which should be same.

For transferring the information in generated images, $\mathcal{L}_{\text{OMGD-KD}}$ is defined as

$$\mathcal{L}_{\text{OMGD-KD}} = \lambda_{\text{SSIM}} \mathcal{L}_{\text{SSIM}} + \lambda_{\text{PL}} \mathcal{L}_{\text{PL}} + \lambda_{\text{recon}} \mathbb{E}_{\boldsymbol{x}} \| G^s(\boldsymbol{x}; \phi^s) - G^t(\boldsymbol{x}; \phi^t) \|_{1,1}, \tag{18}$$

where $\mathcal{L}_{\text{SSIM}}$ and $\mathcal{L}_{\text{PL}}$ denote the structural similarity loss (SSIM) [12] and the perceptual loss [13], respectively, while λ_{SSIM}, λ_{PL}, and λ_{recon} are hyperparameters. In the above equation, $\| \cdot \|_{1,1}$ is an operator to sum the absolute values of all elements.

A.2.2 GCC [5]

For GCC, $\mathcal{L}_{\text{algo}}$ is defined as follows:

$$\mathcal{L}_{\text{algo}} = \mathcal{L}_{\text{GAN}} + \mathcal{L}_{\text{GCC-KD}}, \tag{19}$$

where $\mathcal{L}_{\text{GAN}}$ is the adversarial loss for generators and discriminators while $\mathcal{L}_{\text{GCC-KD}}$ is a distillation loss for final and intermediate outputs, which is given by

$$\begin{aligned} \mathcal{L}_{\text{GCC-KD}} = & \sum_{k=1}^{K} \mathbb{E}_{\boldsymbol{x}} \left[d(D_k^t(G^s(\boldsymbol{x}; \phi^s); \psi^t), D_k^t(G^t(\boldsymbol{x}; \phi^t); \psi^t)) \right] \\ & + \sum_{\ell=1}^{L} \mathbb{E}_{\boldsymbol{x}} \left[d(f_\ell(G_\ell^s(\boldsymbol{x}; \phi^s)), G_\ell^t(\boldsymbol{x}; \phi^t)) \right] + \lambda_{\text{recon}} \mathbb{E}_{\boldsymbol{x}} \| G^s(\boldsymbol{x}; \phi^s) - G^t(\boldsymbol{x}; \phi^t) \|_{1,1}. \end{aligned} \tag{20}$$

Here, $D_k^t(\cdot; \psi^t)$ is the intermediate feature map of the k^{th} layer in the teacher discriminator, and $d(\cdot, \cdot)$ is given by

$$d(X, Y) = \lambda_{\text{MSE}} \| X - Y \|_F^2 + \lambda_{\text{style}} \| \text{Gram}(X) - \text{Gram}(Y) \|_F^2, \tag{21}$$

where λ_{MSE} and λ_{style} are hyperparameters and $\| \cdot \|_F$ denotes the Frobenius norm.

A.2.3 GAN-Compression [1]

$\mathcal{L}_{\text{algo}}$ is defined as follows:

$$\mathcal{L}_{\text{algo}} = \mathcal{L}_{\text{GAN}} + \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{distill}} \mathcal{L}_{\text{distill}}, \tag{22}$$

where λ_{recon} and λ_{distill} are hyperparameters while $\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{distill}}$ are defined as

$$\mathcal{L}_{\text{recon}} = \begin{cases} \mathbb{E}_{\boldsymbol{x}, \boldsymbol{y}} \| G^s(\boldsymbol{x}; \phi^s) - \boldsymbol{y} \|_{1,1} & \text{for paired datasets,} \\ \mathbb{E}_{\boldsymbol{x}} \| G^s(\boldsymbol{x}; \phi^s) - G^t(\boldsymbol{x}; \phi^t) \|_{1,1} & \text{for unpaired datasets,} \end{cases} \tag{23}$$

$$\mathcal{L}_{\text{distill}} = \sum_{\ell=1}^{L} \mathbb{E}_{\boldsymbol{x}} \| f_\ell(G_\ell^s(\boldsymbol{x}; \phi^s)) - G_\ell^t(\boldsymbol{x}; \phi^t) \|_F^2. \tag{24}$$

In the above equation, $f_\ell(\cdot)$ indicates the 1×1 convolution operator to match the dimensionality of the two feature maps while $\boldsymbol{y}$ is the ground-truth output.

A.2.4 CAT [2]

$\mathcal{L}_{\text{algo}}$ is defined as follows:

$$\mathcal{L}_{\text{algo}} = \mathcal{L}_{\text{GAN}} + \lambda_{\text{recon}}\mathcal{L}_{\text{recon}} + \lambda_{\text{KA}}\mathcal{L}_{\text{KA}}, \tag{25}$$

where λ_{recon} and λ_{KA} are hyperparameters. In the above equation, $\mathcal{L}_{\text{KA}}$ is given by

$$\mathcal{L}_{\text{KA}} = -\sum_{\ell=1}^{L} \mathbb{E}_{\boldsymbol{x}} \left[\frac{\|\rho(G_\ell^s(\boldsymbol{x};\phi^s))^{\mathrm{T}}\rho(G_\ell^t(\boldsymbol{x};\phi^t))\|_F^2}{\|\rho(G_\ell^s(\boldsymbol{x};\phi^s))^{\mathrm{T}}\rho(G_\ell^s(\boldsymbol{x};\phi^s))\|_F \|\rho(G_\ell^t(\boldsymbol{x};\phi^t))^{\mathrm{T}}\rho(G_\ell^t(\boldsymbol{x};\phi^t))\|_F} \right], \tag{26}$$

where $\rho(\cdot)$ is a reshape operator from a 4D tensor ($\in \mathbb{R}^{n\times c\times h\times w}$) into a 2D matrix ($\in \mathbb{R}^{n\times chw}$).

A.2.5 CAGC [3]

$\mathcal{L}_{\text{algo}}$ is defined as follows:

$$\mathcal{L}_{\text{algo}} = \mathcal{L}_{\text{GAN}} + \lambda_{\text{recon}}\mathcal{L}_{\text{recon}}^{\text{mask}} + \lambda_{\text{distill}}\mathcal{L}_{\text{distill}}^{\text{mask}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}^{\text{mask}}, \tag{27}$$

where λ_{recon}, λ_{distill}, and λ_{LPIPS} are hyperparameters while $\mathcal{L}_{\text{recon}}^{\text{mask}}$, $\mathcal{L}_{\text{distill}}^{\text{mask}}$, and $\mathcal{L}_{\text{LPIPS}}^{\text{mask}}$ are given by

$$\mathcal{L}_{\text{recon}}^{\text{mask}} = \mathbb{E}_{\boldsymbol{x}}\|M \odot (G^s(\boldsymbol{x};\phi^s) - G^t(\boldsymbol{x};\phi^t))\|_{1,1}, \tag{28}$$

$$\mathcal{L}_{\text{distill}}^{\text{mask}} = \sum_{\ell=1}^{L} \mathbb{E}_{\boldsymbol{x}}\|M \odot (f_\ell(G_\ell^s(\boldsymbol{x};\phi^s)) - G_\ell^t(\boldsymbol{x};\phi^t))\|_{1,1}, \tag{29}$$

$$\mathcal{L}_{\text{LPIPS}}^{\text{mask}} = \mathbb{E}_{\boldsymbol{x}}\left[\text{LPIPS}(M \odot G^s(\boldsymbol{x};\phi^s), M \odot G^t(\boldsymbol{x};\phi^t))\right]. \tag{30}$$

Note that M is a binary mask and represents whether the corresponding pixel is located at the object of interest while LPIPS$(\cdot,\cdot)$ [14] measures the perceptual difference between the two input patches.

A.3 More Implementation Details

Figure 4 illustrates the architecture design of the energy-based model in the form of a convolutional neural network. In Figure 4, we set C, the number of channels in intermediate feature maps, to 8 for the Horse $\rightarrow$ Zebra dataset while setting it to 32 for other datasets. "ConvBlock" consists of a convolution with a kernel size of 3 and a LeakyReLU activation function while "ResBlock" is composed of two convolutional layers with a kernel size of 3 and a LeakyReLU together with a residual connection [15].

Following [16], we incorporate spectral normalizations [17] to the weights in all convolutional and linear layers in order to alleviate sharp gradient changes in the energy-based models. In addition, we minimize the squared output from the energy-based models as a regularization term since the output is not bounded, which may also cause training instability. Without the regularization term, we empirically observe that the output becomes numerically unstable. All of these techniques are helpful to improve the training stability.

A.4 Limitation

VEM incurs an extra training cost because it requires to additional short-run MCMC steps. However, the proposed approach quantitatively and qualitatively improves the performance when combined with existing GAN compression approaches.

A.5 Additional Qualitative Results

We present more qualitative results of VEM and the state-of-the-art methods including the original model in Figure 5, 6, 7, 8, 9, 10, 11, and 12 to demonstrate the effectiveness of VEM.

B Code

We attached the source code to facilitate understanding and reproduction of our algorithm. Please check "README.md" in the supplementary material folder which contains how to run VEM.

Concatenate two inputs
3×3 Avg Pooling (stride $= 2$)
ConvBlock, C
ResBlock, $2 \times C$
ResBlock, $4 \times C$
ResBlock, $4 \times C$
ResBlock, $4 \times C$
ResBlock, $4 \times C$
ResBlock, $4 \times C$
ResBlock, $4 \times C$
ReLU Activation
Global Average Pooling (GAP)
Linear layer, 1

Figure 4: Architecture design of the proposed energy-based model.

Table 6: Selected values of λ_{MI} for VEM.

Model	Dataset	Method	λ_{MI}
Pix2Pix	Edges $\rightarrow$ Shoes	OMGD [4] + VEM (Ours)	0.100
	Cityscapes	OMGD [4] + VEM (Ours)	0.200
CycleGAN	Horse $\rightarrow$ Zebra	OMGD [4] + VEM (Ours)	0.100
		CAT [2] + VEM (Ours)	0.005
		GAN-Compression [1] + VEM (Ours)	0.100
	Summer $\rightarrow$ Winter	OMGD [4] + VEM (Ours)	0.100
SAGAN	CelebA	GCC [5] + VEM (Ours)	0.100
StyleGAN2	FFHQ	CAGC [3] + VEM (Ours)	0.050

Figure 5: Qualitative results of Pix2Pix on the Cityscapes dataset. “Original” represents the images generated by the uncompressed Pix2Pix.

Figure 6: Qualitative results of Pix2Pix on the Edges → Shoes dataset. "Original" represents the images generated by the uncompressed Pix2Pix.

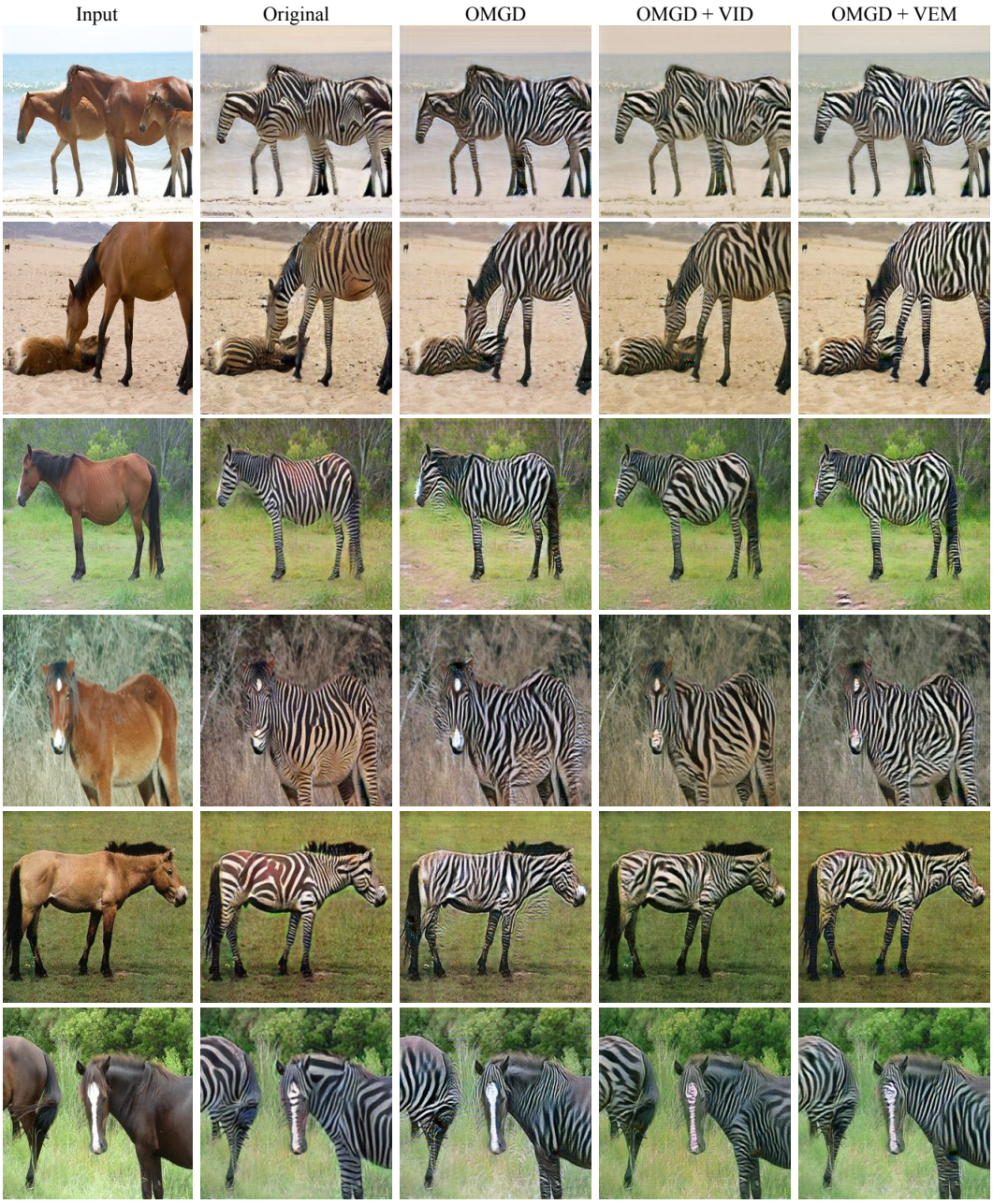

Figure 7: Qualitative results of CycleGAN on the Horse $\rightarrow$ Zebra dataset. "Original" represents the images generated by the uncompressed CycleGAN.

Figure 8: Qualitative results of CycleGAN on the Horse → Zebra dataset. "Original" represents the images generated by the uncompressed CycleGAN.

Figure 9: Qualitative results of CycleGAN on the Horse $\rightarrow$ Zebra dataset. "Original" represents the images generated by the uncompressed CycleGAN.

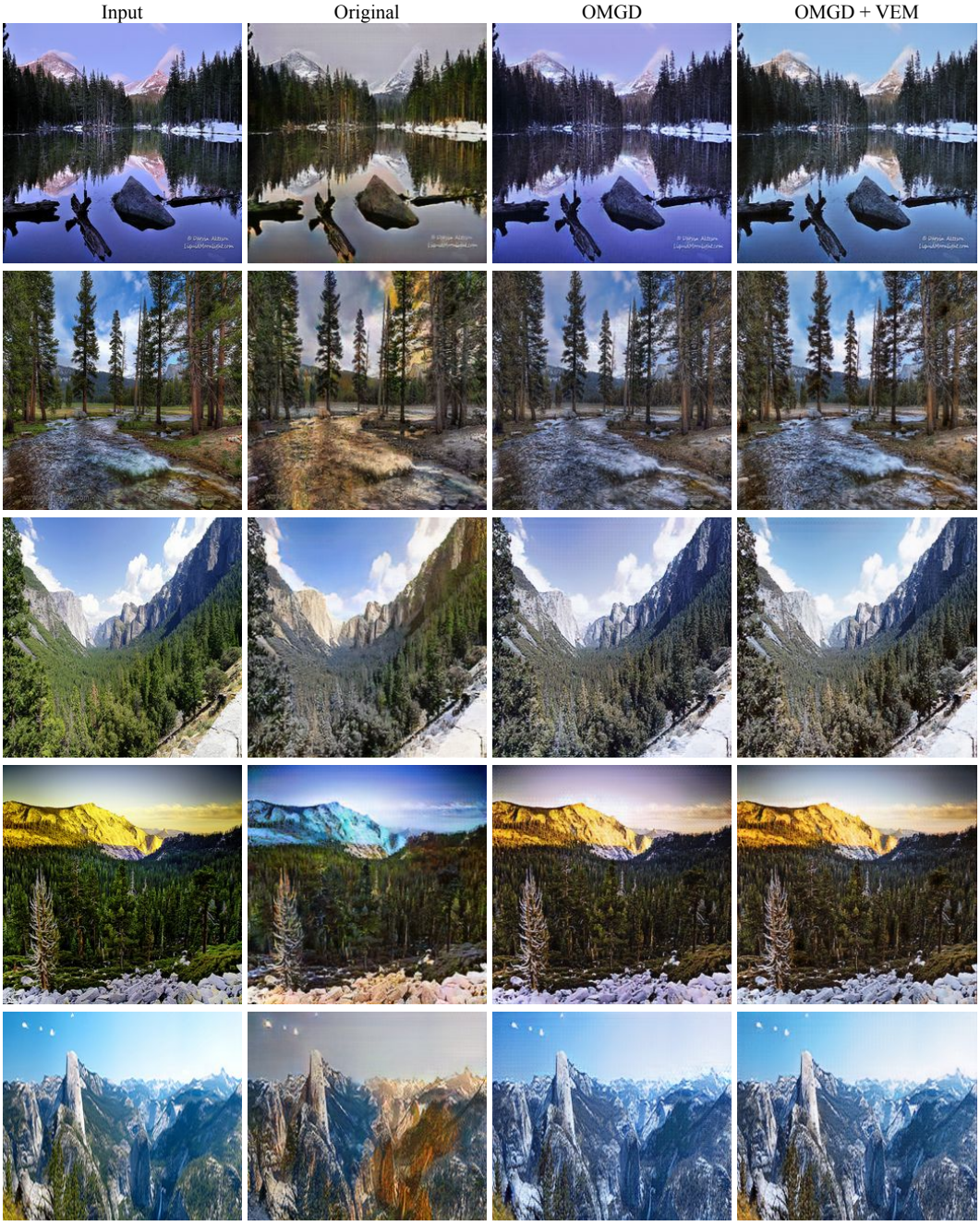

Figure 10: Qualitative results of CycleGAN on the Summer $\rightarrow$ Winter dataset. "Original" represents the images generated by the uncompressed CycleGAN.

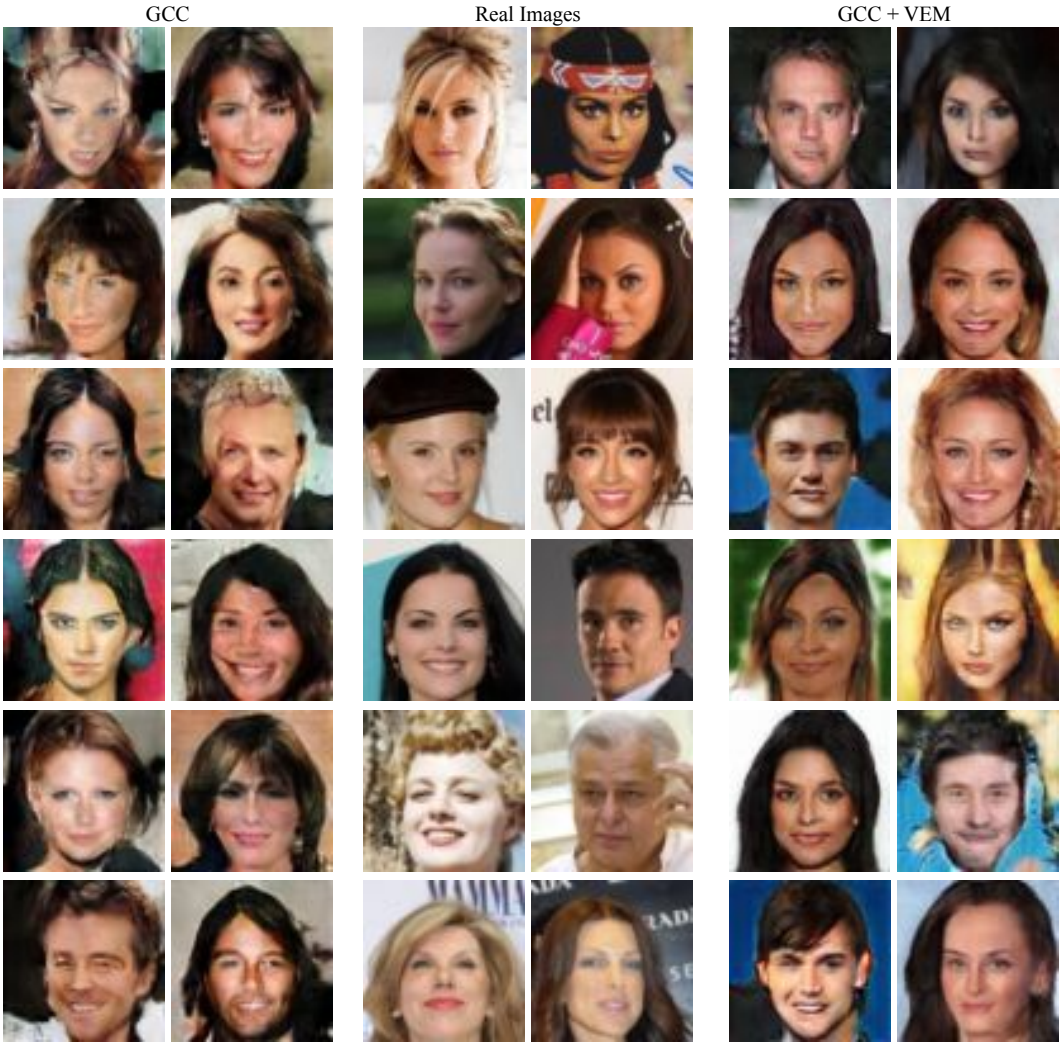

Figure 11: Qualitative results of SAGAN on the CelebA dataset. “Real Images” represents the data samples.

Figure 12: Qualitative results of StyleGAN2 on the FFHQ dataset. “Original” represents the images generated by the uncompressed StyleGAN2.

References

[1] Li, M., Lin, J., Ding, Y., Liu, Z., Zhu, J.Y., Han, S.: Gan Compression: Efficient Architectures for Interactive Conditional GANs. In CVPR. (2020)

[2] Jin, Q., Ren, J., Woodford, O.J., Wang, J., Yuan, G., Wang, Y., Tulyakov, S.: Teachers Do More Than Teach: Compressing Image-to-Image Models. In CVPR. (2021)

[3] Liu, Y., Shu, Z., Li, Y., Lin, Z., Perazzi, F., Kung, S.Y.: Content-Aware GAN Compression. In CVPR. (2021)

[4] Ren, Y., Wu, J., Xiao, X., Yang, J.: Online Multi-Granularity Distillation for GAN Compression. In ICCV. (2021)

[5] Li, S., Wu, J., Xiao, X., Chao, F., Mao, X., Ji, R.: Revisiting Discriminator in GAN Compression: A Generator-discriminator Cooperative Compression Scheme. In NeurIPS. (2021)

[6] Xie, J., Lu, Y., Gao, R., Wu, Y.N.: Cooperative Learning of Energy-based Model and Latent Variable Model via MCMC Teaching. In AAAI. (2018)

[7] Xie, J., Lu, Y., Gao, R., Zhu, S.C., Wu, Y.N.: Cooperative training of descriptor and generator networks. TPAMI (2018)

[8] Xie, J., Zheng, Z., Li, P.: Learning energy-based model with variational auto-encoder as amortized sampler. In AAAI. (2021)

[9] Xie, J., Zhu, Y., Li, J., Li, P.: A tale of two flows: Cooperative learning of Langevin flow and normalizing flow toward energy-based model. In ICLR. (2022)

[10] Hu, J., Shen, L., Sun, G.: Squeeze-and-Excitation Networks. In CVPR. (2018)

[11] Zhou, Z., Zhuge, C., Guan, X., Liu, W.: Channel Distillation: Channel-Wise Attention for Knowledge Distillation. arXiv preprint arXiv:2006.01683 (2020)

[12] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: from Error Visibility to Structural Similarity. TIP (2004)

[13] Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In ECCV. (2016)

[14] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In CVPR. (2018)

[15] He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In CVPR. (2016)

[16] Du, Y., Mordatch, I.: Implicit Generation and Modeling with Energy Based Models. In NeurIPS. (2019)

[17] Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral Normalization for Generative Adversarial Networks. In ICLR. (2018)