Escaping from Collapsing Modes in a Constrained Space

Chia-Che Chang*[1], Chieh Hubert Lin*[1], Che-Rung Lee[1]
Da-Cheng Juan[2], Wei Wei[2], and Hwann-Tzong Chen[1]

[1]Department of Computer Science,
National Tsing Hua University
{chang810249,hubert052702,cherung}@gmail.com,
htchen@cs.nthu.edu.tw,
[2]Google AI, Mountain View, CA, USA
{dacheng,wewei}@google.com

Abstract. Generative adversarial networks (GANs) often suffer from unpredictable mode-collapsing during training. We study the issue of mode collapse of Boundary Equilibrium Generative Adversarial Network (BEGAN), which is one of the state-of-the-art generative models. Despite its potential of generating high-quality images, we find that BEGAN tends to collapse at some modes after a period of training. We propose a new model, called *BEGAN with a Constrained Space* (BEGAN-CS), which includes a latent-space constraint in the loss function. We show that BEGAN-CS can significantly improve training stability and suppress mode collapse without either increasing the model complexity or degrading the image quality. Further, we visualize the distribution of latent vectors to elucidate the effect of latent-space constraint. The experimental results show that our method has additional advantages of being able to train on small datasets and to generate images similar to a given real image yet with variations of designated attributes on-the-fly.

1 Introduction

The main goal of this paper is to provide new insights into the problem of mode collapse in training Generative Adversarial Networks (GANs) [9]. GANs have shown great potential in generating new data based on real samples and have been applied to various vision tasks [4, 6, 10, 19, 20, 26, 27, 29]. Our study points out a simple but effective approach that can be used to improve the stability of training GANs for generating high-quality images with respect to disentangled representations.

GANs comprise two core components: generator G and discriminator D. The two components are optimized with respect to two spaces. One is the latent space Z for the generator, and the other is the data space X associated with

* Indicates equal contribution.

a real data distribution $p_{\text{real}}(x)$ for training data $x \in X$. The objective of the generator is to find a mapping $G : Z \to X$ that maximizes the probability of the discriminator mistakenly accepting a generated image $G(z), z \in Z$ as from $p_{\text{real}}(x)$. On the contrary, the discriminator's objective is to distinguish whether any given $x \in X$ belongs to $p_{\text{real}}(x)$. During training, the generator only learns from the information provided by the discriminator, and aims to estimate a good mapping such that $p_{\text{model}}(G(z))$ is similar to $p_{\text{real}}(x)$.

Compared with auto-encoders [13], GANs can generate sharper images owing to the adversarial loss. However, a downside of adopting the adversarial loss is that it makes the training of GANs unstable. The performance is strongly dependent on hyper-parameters selection, and the generated images tend to have weaker structural coherence.

Boundary Equilibrium Generative Adversarial Network (BEGAN) [3] introduced by Berthelot *et al.* suggests several modifications on the architecture and loss designs, which significantly improve the quality of generated images and the training stability. Another contribution of BEGAN is providing an approximation of convergence for the class of energy-based GANs.

Despite the promising improvements of BEGAN, we empirically observe that BEGAN still unavoidably runs into mode collapses after certain epochs of training. In the meanwhile, neither the approximation of convergence nor the loss functions of BEGAN is able to detect the sudden mode collapses. In our experiments, the exact time when mode collapsing happens is highly related to target image resolution and dataset size. In addition to the typical drawbacks of mode collapsing, this unpredictable behavior also makes BEGAN's intended contribution to providing "global measure of convergence" incomplete.

1.1 Contributions

We propose a new constraint loss toward addressing the mode collapsing problem. We find that the mode-collapsing problem is suppressed after adding the constraint loss. This new loss term does not increase model complexity and is computationally low-cost. Furthermore, it does not introduce any trade-off regarding image quality and diversity. The proposed model is called *BEGAN with a Constrained Space* (BEGAN-CS).

We visualize the latent vectors produced in training phase using Principal Component Analysis (PCA) [1]. In section 3.1, we analyze the effect of the constraint loss and explain why this loss term makes training process stable.

Since BEGAN-CS is more stable during training, it performs consistently well even when the size of training dataset is ten-times smaller than the normal setting, in which BEGAN fails to obtain acceptable results. In section 4.3, our experiment shows that the proposed BEGAN-CS can eventually converge to a better state, while BEGAN ends up at mode collapsing in an early stage.

We further discover that BEGAN is able to learn strong and high-quality disentangled representations in an unsupervised setting. The learned disentangled representations could be used to modify the underlying attributes of generated images. In the meanwhile, owing to the constraint loss, BEGAN-CS can accomplish approximation $Enc(x^*) \simeq z^*$ on-the-fly for any given real image x^*, where $G(z^*)$ is an approximate image to x^* under the fixed generator weights. By leveraging the z^* approximation and the disentangled representations, BEGAN-CS can generate on the fly a set of images conditioning on a real image x^*. The generated images are visually similar to the given real image and are able to exhibit the adjustable disentangled attributes.

2 Related Work

Deep Convolutional Generative Adversarial Network (DCGAN) [24] improves the original GAN [9] by employing a convolutional architecture to achieve better stability of training and enhanced quality of generated images. Salimans *et al*. further present several practical techniques for training GANs [25]. Nevertheless, avoiding mode collapsing while keeping the quality of generated images is still a challenging issue in practice.

Energy-Based Generative Adversarial Network (EBGAN) [30] introduces another perspective for formulating GANs. EBGAN implements the discriminator as an auto-encoder with per-pixel error. Boundary Equilibrium Generative Adversarial Network (BEGAN) [3] shares the same discriminator setting as EBGAN and makes several improvements on the designs of architecture and loss function. One of BEGAN's core contributions is introducing the equilibrium concept, which balances the power between the generator and the discriminator. With these improvements, BEGAN provides fast and stable training convergence, and is capable of generating high visual-quality images. Another contribution of BEGAN is providing an approximate measure of convergence. The earlier class of GANs lacks convergence measurement. Not until later, a new class of GANs exemplified by Wasserstein Generative Adversarial Network (WGAN) [2] introduces a new loss metric, which correlates with the generator's convergence. To our knowledge, BEGAN yields an alternative class of GANs that also has a loss correlated with convergence measurement.

Apart from the class of energy-based GANs, Progressive Growing of Generative Adversarial Networks (PGGANs) [14] is another approach to generating high-quality images. By changing the training procedure without modifying the original GAN loss, PGGANs are able to increase training stability and to produce diverse yet high-resolution (up to 1024×1024 pixels) images.

The z^* approximation property of BEGAN-CS is similar to another class of bijective GANs, which constructs a bijection between the latent space Z and the data space X. This class of models includes ALI [8], BiGAN [7], VEEGAN [28] and [15]. These four methods share a similar characteristic, requiring additional effort to optimize an extended network. VEEGAN introduces an extra reconstructor network F_θ, which maps real data distribution $p(x)$ to a Gaussian. ALI/BiGAN both introduce an additional encoder network in the generator, and try to build up a bijection function. For [15], the loss term L_s (Eq. (9) in [15]) has a pre-requirement that the generator must include the real images in its latent space. They introduce an extra encoder network in generator to fulfill this requirement.

In comparison, BEGAN-CS introduces a light-weight loss that utilizes the built-in mechanism of BEGAN without a need of extra networks. This makes the latent space inverting function jointly optimizable with the discriminator. Also, the constraint loss is a very strong indicator, detecting and protecting the model from mode collapsing. We also include further experimental comparisons with the class of bijective GANs in section 4.6.

3 Methods

Mode collapse is a phenomenon that the generated images get stuck in or oscillate between a few modes. This phenomenon under BEGAN's setting has a unique characteristic. Since every sample shares the same encoder in the discriminator of BEGAN, the generated images that collapse at the same mode will share similar latent vectors as encoded by the encoder.

By leveraging this property, we propose the *latent-space constraint loss* ($\mathcal{L}_c$), or the *constraint loss* for short. It constrains the norm of the difference between the latent vector z and the internal state of encoder $Enc(G(z))$, where Enc is the encoder within the discriminator. During the training process, the constraint loss is only optimized with respect to the discriminator. Although the mode-collapsing problem happens on the generator side, adding the constraint loss directly to the generator would expose too much information to the generator about how to exploit the discriminator, and thus turns out accelerating the occurrence of mode collapse. The constraint loss can also be viewed as a regularizer, which guides the function $Enc(G(\cdot))$ to be an identity function, and forces the encoder of the discriminator to retain the diversity and uniformity of randomly sampled $z \in Z$.

Fig. 1 is an overview of the full-architecture of BEGAN-CS. The objective function of BEGAN-CS is mostly similar to BEGAN, except the additional constraint loss. The full objective of BEGAN-CS includes

$$\mathcal{L}_G = \mathcal{L}(G(z_G; \theta_G); \theta_D)\,, \quad \text{for } \theta_G \tag{1}$$

and

$$\mathcal{L}_D = \mathcal{L}(x_{\text{real}}; \theta_D) - k_t \cdot \mathcal{L}(G(z_D; \theta_G); \theta_D) + \alpha \cdot \mathcal{L}_c\,, \quad \text{for } \theta_D \tag{2}$$

with

$$\begin{cases} \mathcal{L}_c = \| z_D - Enc(G(z_D)) \|\,, & \text{(the constraint loss)} \\ k_{t+1} = k_t + \lambda(\gamma \mathcal{L}(x; \theta_D) - \mathcal{L}(G(z_G; \theta_G); \theta_D))\,, & \text{for each epoch}\,. \end{cases} \tag{3}$$

The total loss $\mathcal{L}_G$ of the generator and the total loss $\mathcal{L}_D$ of the discriminator are optimized to solve for the parameters θ_G and θ_D, respectively. The function $\mathcal{L}(x; \theta_D) = \|x - D(x)\|$ associated with θ_D computes the norm of the difference between any given image x and its reconstructed image $D(x)$ by the decoder of

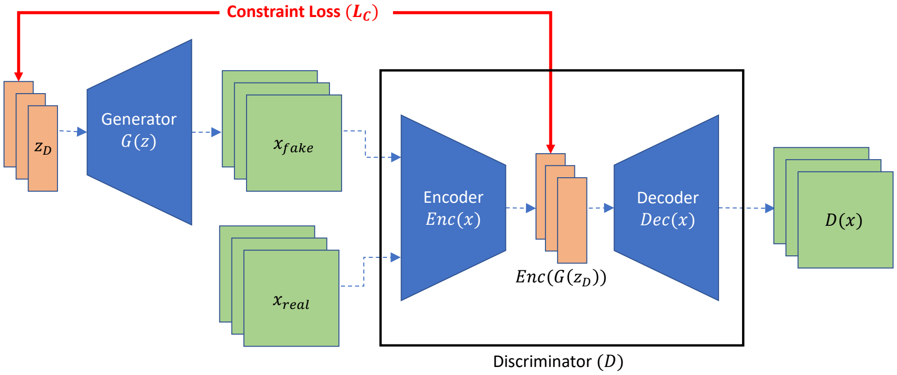


Fig. 1: An overview of BEGAN-CS.

the discriminator. The latent vectors z_D and z_G are randomly sampled from Z. The variable $k_t \in [0,1]$ controls how much emphasis to put on $\mathcal{L}(G(z_D;\theta_G);\theta_D)$. The hyper-parameter $\gamma \in [0,1]$ balances between the real-image reconstruction loss $\mathcal{L}(x;\theta_D)$ and the generated-image discrimination loss $\mathcal{L}(G(z_G;\theta_G);\theta_D)$. The hyper-parameter α is a weighting factor for constraint loss. The constraint loss $\mathcal{L}_c$ is to enforce $Enc(G(\cdot))$ to be an identity function for z_D.

3.1 Latent Space Analysis

For further illustrating the effectiveness of our method and analyzing the root cause of mode collapsing, we visualize the latent space through time with and without the constraint loss. We take PCA as our choice of dimensionality reduction method, and project the latent vectors onto two-dimensional space. Another common choice of dimensionality reduction for visualization is t-Distributed Stochastic Neighbor Embedding (t-SNE) [22]. For the latent space, we are more interested in the density and distribution of the points rather than the relative nearness between points or clusters. As a result, PCA is more suitable for our analysis.

Fig. 2 shows a preliminary analysis of BEGAN and BEGAN-CS. We train both models on the CelebA dataset [21]. The 64-dimensional latent vectors of generated images ($Enc(G(z))$) and real images ($Enc(x)$) are projected onto two-dimensional space via PCA.

In this experiment, BEGAN gets into mode collapse at epoch 23. In addition to the obvious change in the shape of distribution after BEGAN mode-collapsing, our empirical analysis also shows two strong patterns. First, in comparison with BEGAN, the latent-vector distribution (in red) of images generated by BEGAN-CS can better fit the real images' latent-vector distribution (in blue). The latent vectors of BEGAN-CS scatter more uniformly across all epochs.

Second, for BEGAN without adding the constraint loss, both the variance of real images' latent vectors (Var(real)) and the variance of generated images' latent vectors (Var(gen)) grow rapidly as the number of epochs increases. Our hypothesis is that the latent spaces of real images and generated images both expand too rapidly and non-uniformly. Since the number of training data is fixed, as the latent space of real images expands, the density of real images decreases. In the end, the generator of BEGAN reaches a low-density area in the latent space where there is only a few latent vectors of real images nearby. The generator of BEGAN then gets stuck in that area. In contrast, BEGAN-CS has the latent-space constraint as a regularizer, which restricts the latent spaces of real images and generated images expand incautiously. In other words, the constraint loss limits the distribution of $Enc(G(z))$ to be similar to uniform distribution.

3.2 Obtaining Optimal z^* in One-Shot

Given an image x^*, finding an optimal latent vector z^* such that $\|G(z^*)-x^*\| < \epsilon$ for some small ϵ is a challenging problem for GANs. Traditionally, z^* can be obtained by back-propagation for solving the optimization $\min_{z^*}(\|G(z^*) - x^*\|)$. We name this optimization process as z^*-search. However, z^*-search is time-consuming and needs to run for each inference individually, and thus is impractical for real-world applications.

In the case of BEGAN-CS, the constraint loss works as a regularizer, guiding the composite function $Enc(G(z)) \simeq z$ to be similar to an identity function. Consider the definition of z^*, where $G(z^*) = x^*$. We know that $Enc(G(z^*))$ should be close to z^* due to the identity property. This implies that we may take x^* and obtain $Enc(x^*)$ as an approximation to z^* after a single pass through the encoder $Enc(x^*)$.

3.3 Disentangled Representation Learning and Application

We find that BEGAN is able to learn strong and high-quality disentangled representations in an unsupervised setting. The direction of any vector within latent space Z has a universally meaningful semantic, such as mixture of gender, age, smile and hair-style. These learned representations can be combined with vector arithmetic operations to generate images with multiple designated representations.

However, these disentangled representations are only effective for latent vectors, which is a strong restriction that forbids many GAN models to use the disentangled representation for practical applications, since obtaining the latent vectors via z^*-search is computation-demanding. In the meanwhile, as we have shown in section 3.2, BEGAN-CS is able to produce the approximation of z^* on-the-fly. By adding multiple selected representation vectors to the approximated z^* with respect to any given real image x^*, we can generate images that are visually similar to x^* and comprise the selected representations at the same time. We demonstrate this idea with a real example produced by BEGAN-CS in Fig 3. In this example, the generated image of Fig 3d acquires both hair-styles shown Fig 3b & Fig 3c. For BEGAN, which lacks the ability of estimating z^* directly, the same effect may be forcibly achieved through time-consuming z^*-search to obtain suitable z^*. Unfortunately, z^*-search causes the major bottleneck at inference time and is therefore hard to use in real-world scenarios.

Similar applications can also be achieved using Variational Auto-Encoder (VAE) based models [17, 12, 18] or other task-specific GAN models, such as InfoGAN [5]. However, the images generated by VAE-based models tend to be blurry, while InfoGAN cannot generate high-quality results as BEGAN does. In comparison, our results are more promising in terms of stability and quality.

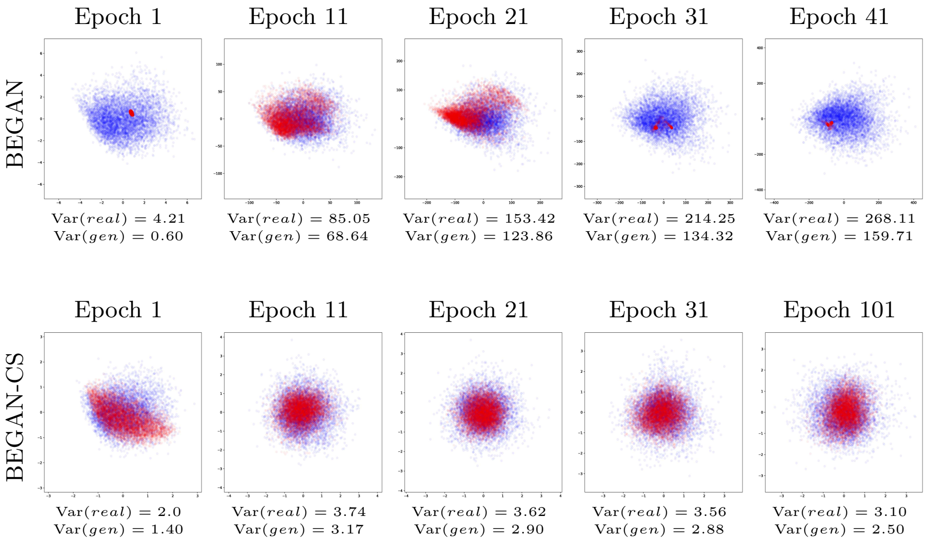


Fig. 2: We visualize the distributions of latent vectors of BEGAN and BEGAN-CS over epochs. Both BEGAN and BEGAN-CS are trained on CelebA dataset under 64×64 resolution and batch size 64. Each graph consists of 6,400 random real images' latent vectors, *i.e.* $Enc(x)$, and 6,400 generated images' latent vectors, *i.e.* $Enc(G(z))$. The upper five graphs are generated by BEGAN, while the bottom five graphs are produced by BEGAN-CS. PCA is performed separately at each epoch based on the latent vectors of the real images. Each blue point represents a latent vector of a real image after applying PCA, and the red points correspond to the latent vectors of the generated images. The text under each graph lists the variance of real images' latent vectors (Var(real)) and the variance of generated images' latent vectors (Var(gen)). During the training of BEGAN, the variances of the distributions of latent vectors keep growing. Note that most of the graphs are created with a fixed interval of 10 epochs, except the bottom-right graph directly skips to the 101st epoch to highlight the effectiveness of BEGAN-CS. BEGAN has already collapsed before the 41st epoch.

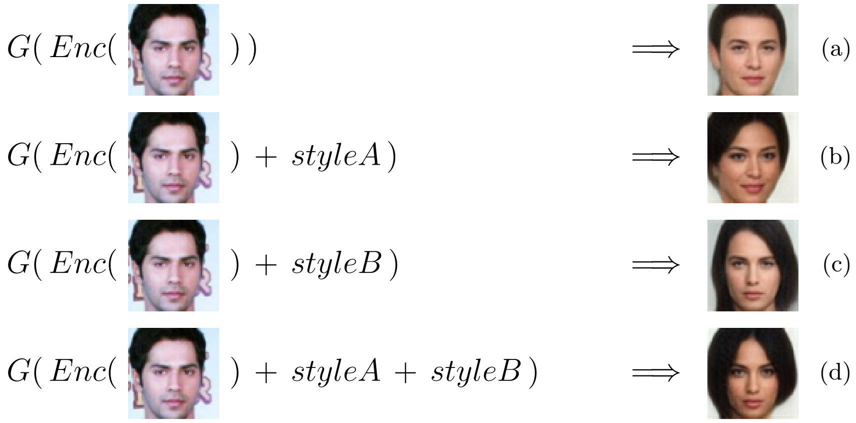


Fig. 3: An example of disentangled representations. The "*styleA*" and "*styleB*" are two learned disentangled representations. Note that these representations are universal and can be applied to any latent vector z for generating images $G(z + style)$ with designated attributes. (a) Approximate z^* by $G(Enc(x^*))$ in one-shot. (b) & (c) The learned disentangled representations can be combined with $G(Enc(x^*))$. (d) Vector arithmetic with multiple disentangled representations. In this case, the generated image has both hair-styles shown in *styleA* and *styleB*.

4 Experiments

We train BEGAN-CS using the CelebA dataset for all the experiments presented in this paper. BEGAN-CS does not adopt the learning rate decay technique described in BEGAN's original paper, since the training process of BEGAN-CS is already very stable. The hyper-parameter α that controls the importance of the constraint loss is set to 0.1 as the default value. We use L2-norm in

$$\mathcal{L}(x; \theta_D) = \|x - D(x)\|$$

throughout the experiments, while in practice, L1-norm can also be used. For any hyper-parameter that is not mentioned, we choose the same value as in BEGAN's original setting.

4.1 Effectiveness of the Constraint Loss

In Fig. 4, we validate the effectiveness of the constraint loss. We show the generated images at specific epochs during the training of BEGAN and BEGAN-CS on the CelebA dataset. The image resolution is 64 × 64 and the batch size is 64. BEGAN-CS can continuously be trained up to 100 epochs without any evidence of mode collapsing, loss of diversity, or reduction in quality. In contrast, BEGAN encounters mode collapse at the 25th epoch (*i.e.*, the time-step B in Fig. 4). In addition to the advantage of preventing from mode collapse, the proposed BEGAN-CS model also maintains a very good performance in generating high-quality images.

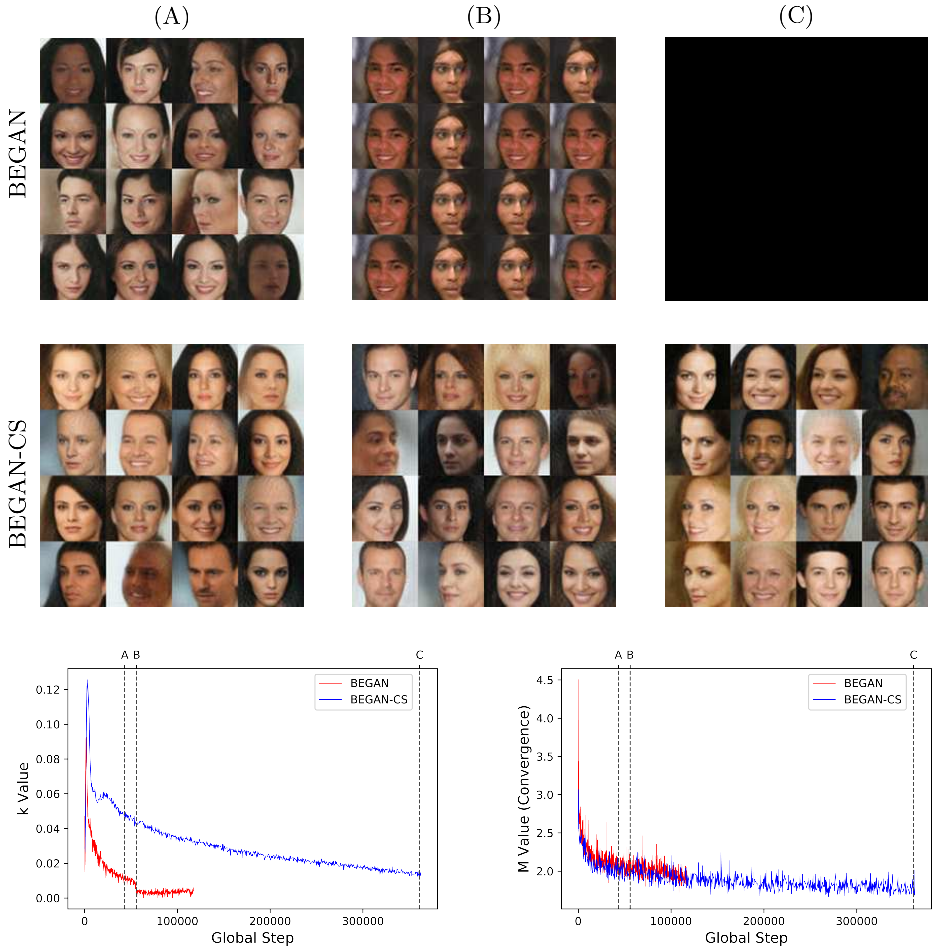


Fig. 4: We validate the effectiveness of the constraint loss by showing the generated images at specific epochs during the training of BEGAN and BEGAN-CS on the CelebA dataset. The image resolution is 64 × 64 and the batch size is 64. Note that BEGAN fails to reach epoch C since it already collapses at epoch B. In contrast, BEGAN-CS survives after epoch C. Furthermore, BEGAN-CS maintains a very good performance in generating high-quality images.

4.2 Observing the Sudden Mode Collapsing

An interesting finding during our experiments is the timing of mode collapsing. As is mentioned in [3], the global measure of convergence can be used by BEGAN to determine whether the network has reached the final state or if the model has collapsed. However, in practice we are not able to observe significant evidence of mode collapsing directly from the value of the convergence measure. Instead, the evidence of mode collapse are more often to be observed from the k value. The k value in BEGAN controls how much attention is paid on $\mathcal{L}(G(z))$. According to our observation, every time the k value suddenly drops, BEGAN is going to collapse shortly.

4.3 Better Convergence on Small Datasets

The dataset size is also an important factor for the timing of mode collapse. Under a setting of reducing the training dataset CelebA to 1/10 of its original size, BEGAN collapses earlier than training on full dataset. The early occurrence of mode collapse keeps BEGAN from converging to an optimal state. The timestep A in Fig. 6 is the best state that BEGAN can achieve during its training on the down-sized CelebA dataset. On the other hand, BEGAN-CS has a more stable training process. In Fig. 6, BEGAN-CS can continuously optimize on the 1/10 down-sized CelebA dataset without encountering mode collapse, and eventually converges to a better state than BEGAN.

4.4 FID Score Curve Comparison

For the quantitative comparison to demonstrate the effectiveness of the proposed constraint loss, we accordingly report "Fréchet Inception Distance" (FID) [11] score through time of BEGAN and BEGAN-CS in Fig. 5. The experiments are conducted at 64 × 64 resolution. It can be seen in Fig. 5 that, during training, the FID of BEGAN-CS does not increase drastically as BEGAN.

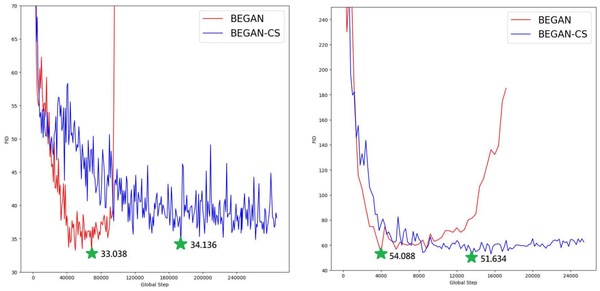


Fig. 5: FID through time. (Left) Full CelebA. (Right) 1/10 CelebA.

4.5 Obtaining Optimal z^* in One-Shot

In section 3.2, we have shown that BEGAN-CS can approximate optimal z^* with $Enc(x^*)$. Appendix A shows the experimental results of interpolation with obtained z^* from z^*-search using different GAN architectures. The experiments may serve as proofs of concept for comparing the well-known GANs architectures, including FisherGAN [23], PGGAN [14], and BEGAN. The experimental results show that the obtained $G(Enc(x^*))$ of BEGAN-CS is visually similar to x^*. In contrast, the original BEGAN and other state-of-the-art GANs require time-consuming z^*-search for 10,000 iterations to obtain competitive results. It would take 340 seconds to 3,970 seconds depending on the network architecture.

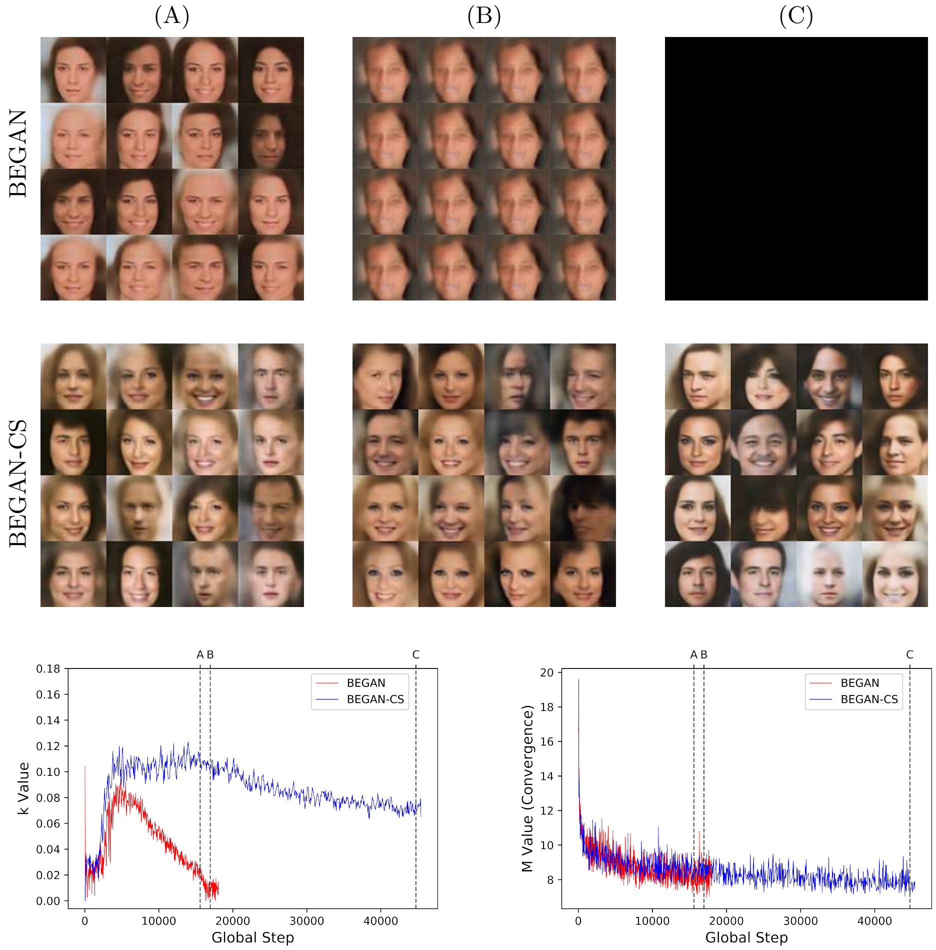
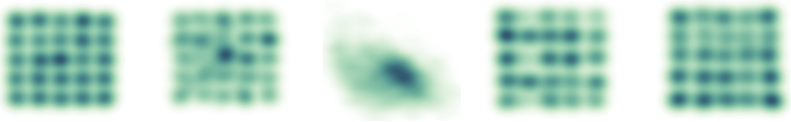


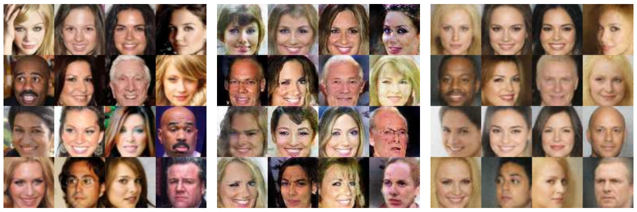
Fig. 6: Better convergence of BEGAN-CS on small datasets. We show the generated images at selected epochs during training BEGAN and BEGAN-CS on a 1/10 sized subset of CelebA. Training images are of 128 × 128 resolution and the batch size is 24. BEGAN-CS is stable and converges to a particularly better state than BEGAN. The best state of BEGAN is at time-step A with degraded quality, while BEGAN-CS can generate higher-quality results at time-step C.

However, the quality of the z^*-search result is still unstable and the searched image frequently looks quite different to the given real image, such as wrong gender or incorrect head pose. More examples on z^*-search with different GAN models and different numbers of optimization iterations are shown in Appendix B.



(a) Real (b) VEEGAN (c) ALI (d) BEGAN (e) BEGAN-CS

Fig. 7: Experimental results on the synthetic dataset introduced by VEEGAN.



(a) Real images (b) ALI (c) BEGAN-CS

Fig. 8: Image reconstruction results.

4.6 Comparison with Bijective Models

VEEGAN runs experiments on a synthetic toy dataset which consists of 25 independent Gaussian distributions, and observes better stable and higher diversity than other GANs. We accordingly run the similar experiment and provide comparisons in Fig. 7 for VEEGAN, ALI, BEGAN, and BEGAN-CS. We find that the vanilla BEGAN can already fit most of the modes of the real data distribution, though it requires extensive hyper-parameters tuning. Furthermore, BEGAN-CS can stabilize the training and converge to a final state of higher quality. Although VEEGAN can fit to all modes, the distribution is relatively blurry and less similar to the real data distribution. Lastly, ALI fails to fit to the real data distribution.

The hyper-parameters we used for BEGAN and BEGAN-CS on the toy dataset are $\alpha = 0.1$, γ=25, λ=1e-4. We use Adam [16] optimizer with lr_d=1e-4, lr_g=5e-4, β_1=0.5 and β_2=0.999. The latent dimension of Z is set to 32. Both the generator and discriminator are consist of 2 layers of feed-forward network with 128 nodes and ReLU activation. We also set the weight initialization function to be a uniform-random sampler in range $[-\sqrt{9/n}, \sqrt{9/n}]$, which n is the number of layer input.

We also present qualitative comparisons on image reconstruction with BEGAN-CS and ALI in Fig. 8. We find that the loss functions used by all three methods, ALI, BiGAN, and BEGAN-CS, do not guarantee that the reconstruction results are identical to the real images. BEGAN-CS is better at retaining some of the important features, such as hair color, skin color, gaze, and head pose.

4.7 On-the-Fly Representation Manipulation

In section 3.3, we demonstrate a new application of BEGAN-CS with the disentangled representations. By obtaining the approximation of z^* with $Enc(x^*)$ and applying the selected disentangled representations, BEGAN-CS can generate images that are visually similar to x^* and exhibit the selected representations at the same time. As a proof of concept, we visualize the process of adding single representation in Fig. 9 and multiple representations in Fig. 10.

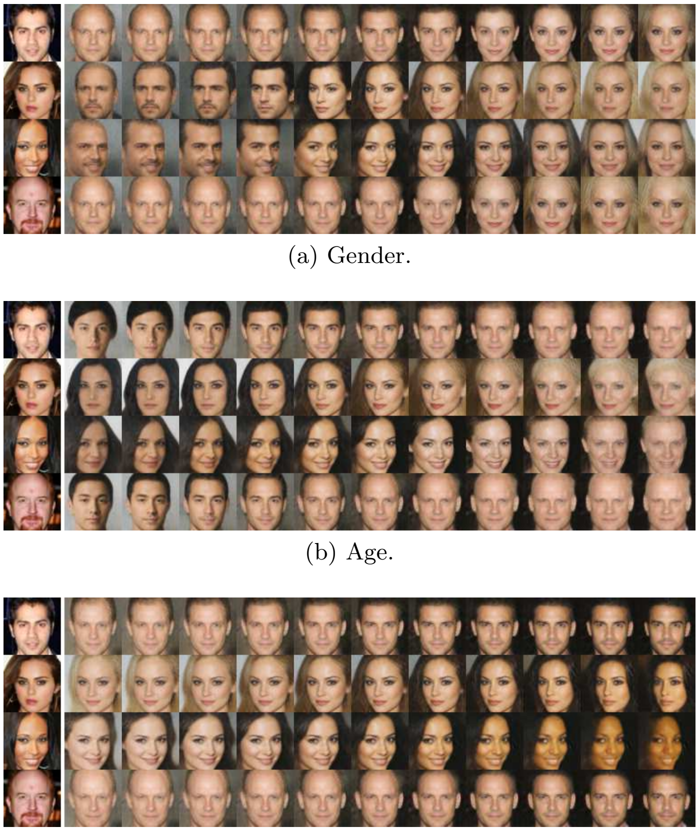
In Fig. 9, we first obtain the approximation of z^* from $Enc(x^*)$. Then for each dimension i, we linearly interpolate and replace the value of latent vector z^* at its ith dimension by a grid value in $[-5, 5]$ with step size 1, and thus can generate a series of images based on the modified latent vectors. The images show that each dimension of the latent space Z represents a universal disentangled representation. We can perform similar visual transformations to any $z \in Z$. Fig. 9 shows some of the interesting disentangled representations. The full visualization across the 64 dimensions is displayed in Appendix C.

The learned disentangled representations can also be used to perform multiple vector arithmetic operations on latent vectors. This property enables us to control multiple attributes of a fixed image at the same time by adjusting multiple dimension values on the corresponding latent vector. We visualize the results of combining two different representations in Fig. 10.

5 Conclusion

We identify that BEGAN suffers from the unpredictable mode-collapsing problem. The precise time when mode collapsing happens is non-deterministic, highly related to the resolution of generated images and the size of training dataset. We propose *BEGAN with a Constrained Space* (BEGAN-CS) toward addressing the mode-collapsing problem and visualize the effect of constraint loss in the latent space. We experimentally show that the model-collapsing problem is suppressed after adding the constraint loss. BEGAN-CS performs particularly better than BEGAN when the size of training dataset is ten-times smaller than the normal setting. These advantages enable the class of energy-based GANs to move on to the next challenge of generating even higher resolution images.

We also discover that BEGAN can learn salient and high-quality disentangled representations in an unsupervised setting. Combined with the particular property that BEGAN-CS is able to approximate z^* on-the-fly, BEGAN-CS can generate images that are visually similar to the given real image and able to exhibit the adjustable disentangled properties. "Obtaining z^* in one-shot" and "adjustable image attributes" are two interesting properties that have various potential applications, such as style manipulation and attribute-based editing.



(a) Gender.

(b) Age.

(c) Hair and skin color.

Fig. 9: Selected disentangled representations produced by BEGAN-CS at 64×64 resolution. For each series of images, the left-most image is the fixed real image x^*. In each sub-figure, we first obtain approximation of z^* using $Enc(x^*)$. For each dimension i, we linearly interpolate and replace the ith dimension of z^* by a value in $[-5, 5]$ with step size 1, and then generate the image set $\{G(z_i^*)\}$.

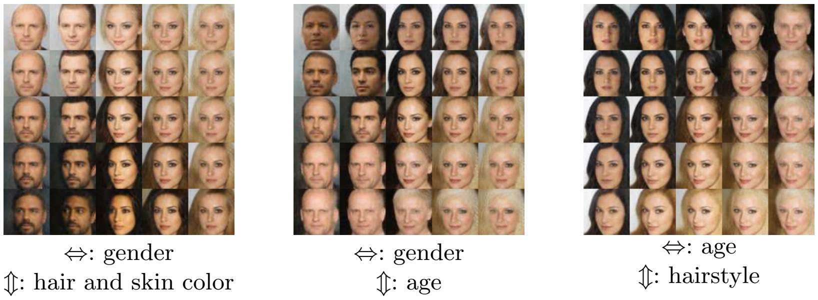


Fig. 10: Two-dimensional combinations of disentangled representations.

References

1. Principal component analysis. Chemometrics and Intelligent Laboratory Systems **2**(1)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. CoRR **abs/1701.07875** (2017)
3. Berthelot, D., Schumm, T., Metz, L.: BEGAN: boundary equilibrium generative adversarial networks. CoRR **abs/1703.10717** (2017)
4. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 95–104 (2017)
5. Chen, X., Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 2172–2180 (2016)
6. Dai, B., Fidler, S., Urtasun, R., Lin, D.: Towards diverse and natural image descriptions via a conditional GAN. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 2989–2998 (2017)
7. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. CoRR **abs/1605.09782** (2016)
8. Dumoulin, V., Belghazi, I., Poole, B., Lamb, A., Arjovsky, M., Mastropietro, O., Courville, A.C.: Adversarially learned inference. CoRR **abs/1606.00704** (2016)
9. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp. 2672–2680 (2014)
10. Gwak, J., Choy, C.B., Garg, A., Chandraker, M., Savarese, S.: Weakly supervised generative adversarial networks for 3d reconstruction. CoRR **abs/1705.10904** (2017)
11. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. pp. 6629–6640 (2017)
12. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework (2016)
13. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. science **313**(5786), 504–507 (2006)
14. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. CoRR **abs/1710.10196** (2017)
15. Khan, S.H., Hayat, M., Barnes, N.: Adversarial training of variational autoencoders for high fidelity image generation. In: 2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018. pp. 1312–1320 (2018)
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2014)

17. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. CoRR **abs/1312.6114** (2013)
18. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. In: Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016. pp. 1558–1566 (2016)
19. Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 105–114 (2017)
20. Li, Y., Liu, S., Yang, J., Yang, M.: Generative face completion. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 5892–5900 (2017)
21. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (2015)
22. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
23. Mroueh, Y., Sercu, T.: Fisher GAN. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. pp. 2510–2520 (2017)
24. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR **abs/1511.06434** (2015)
25. Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 2226–2234 (2016)
26. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 2242–2251 (2017)
27. Souly, N., Spampinato, C., Shah, M.: Semi supervised semantic segmentation using generative adversarial network. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017. pp. 5689–5697 (2017)
28. Srivastava, A., Valkov, L., Russell, C., Gutmann, M.U., Sutton, C.A.: VEEGAN: reducing mode collapse in gans using implicit variational learning. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. pp. 3310–3320 (2017)
29. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 2962–2971 (2017)
30. Zhao, J.J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. CoRR **abs/1609.03126** (2016)