Hindawi
Computational Intelligence and Neuroscience
Volume 2018, Article ID 6465949, 14 pages
https://doi.org/10.1155/2018/6465949

Research Article

Information-Based Boundary Equilibrium Generative Adversarial Networks with Interpretable Representation Learning

Junghoon Hah, Woojin Lee, Jaewook Lee, and Saerom Park

Industrial Engineering, Seoul National University, 1 Gwanakro, Gwanak-gu, Seoul 08826, Republic of Korea

Correspondence should be addressed to Saerom Park; psr6275@snu.ac.kr

Received 30 January 2018; Revised 3 August 2018; Accepted 4 September 2018; Published 17 October 2018

Academic Editor: Antonino Laudani

This paper describes a new image generation algorithm based on generative adversarial network. With an information-theoretic extension to the autoencoder-based discriminator, this new algorithm is able to learn interpretable representations from the input images. Our model not only adversarially minimizes the Wasserstein distance-based losses of the discriminator and generator but also maximizes the mutual information between small subset of the latent variables and the observation. We also train our model with proportional control theory to keep the equilibrium between the discriminator and the generator balanced, and as a result, our generative adversarial network can mitigate the convergence problem. Through the experiments on real images, we validate our proposed method, which can manipulate the generated images as desired by controlling the latent codes of input variables. In addition, the visual qualities of produced images are effectively maintained, and the model can stably converge to the equilibrium. However, our model has a difficulty in learning disentangling factors because our model does not regularize the independence between the interpretable factors. Therefore, in the future, we will develop a generative model that can learn disentangling factors.

1. Introduction

The generative adversarial network (GAN) model that is one of the prominent generative models has been successfully applied to image generation task [1]. The basic idea of GAN is to adversarially train two different models, a generator and a discriminator. The generator aims to generate fake samples similar to real samples from random noise variables. Meanwhile, the discriminator learns to distinguish real samples from fake samples obtained by the generator. When a GAN model reaches the convergence, the generator can give indistinguishable samples with real samples. In addition, distributed representations of the inputs can be obtained from the hidden representations of the GAN model, which reflect the underlying factors of variation that generate the data [2]. However, the original GAN model has difficulties in training and modal collapse [3].

To overcome these problems, many improved models have been proposed [4–6]. While the original GAN model has a binary classification discriminator that distinguishes fake samples from real ones, the recent models construct more informative discriminator costs such as an autoencoder cost that assign lower cost to real samples and higher cost to fake ones [3, 6, 7]. Boundary Equilibrium GAN (BEGAN) [3] used an autoencoder cost as a discriminator cost and balanced adversarial network using proportional control theory, and it, as a result, converged to diverse images of the highest visual quality without a complex alternating training procedure. However, the learned representations of images are entangled and barely interpretable because each dimension in distributed representations does not have a specific meaning.

Unsupervised learning is a general problem, which requires extraction of some valuable information from unlabeled data. Representation learning is one popular framework, which tries to learn a representation from unlabeled data that can be easily decoded [8–10]. In generative models, if we can learn interpretable representations with

methods of unsupervised learning, it can be useful for generating new data. In fact, there are many generative models that construct new data with high quality with arbitrarily bad representations [11]. However, good generative models, including the idea of unsupervised learning, are expected to learn an interpretable representation and synthesize new data that can be manipulated as desired.

On the assumption that a good representation can find the underlying causes from the samples and is interpretable, supervised or semisupervised generative models have been developed [12–17]. Early methods of representation learning were based on autoencoders or RBMs (Restricted Boltzmann Machines) [18, 19]. Recently, VAEs (Variational Autoencoders) [12] achieved splendid semisupervised results on MNIST dataset [20], and GANs learned image representation that enables linear algebra on coded space [4]. There were several attempts to learn disentangled or interpretable representations with supervised datasets [21]. Such methods train the model to match one class of representations and supplied label. Similar to that, adversarial autoencoders [22] and VAEs learned representations with class label separated from other variations. To avoid those methods that explicitly label the variations, weakly supervised methods were introduced.

Because VAE and GAN frameworks have been popularly used in generative modeling, many models for learning disentangled or interpretable representations based on them have been proposed [13, 16, 17]. In generative models, it is important that some factors can be manipulated to generate a new image. However, in supervised models, what disentangling factors are learned is determined when constructing a model. Therefore, the factors that can be manipulated are also determined. Mathieu et al. developed a conditional generative model for learning to disentangle the factors of variation [13]. Their model consisted of two kinds of factors of variations such as the specified factors and the remaining factors. The specified part is modeled as VAE framework, and the remaining part is modeled as GAN framework. Xiao et al. proposed DNA-GAN, a supervised learning model for learning disentangling factors of variation [17]. They iteratively trained the model to address the problem of unbalanced multiattribute datasets. Their model also consisted of attribute-relevant and attribute-irrelevant parts. Higgins et al. developed a Symbol-Concept Association Network (SCAN) which learns higher-level concepts based on disentangled visual primitives, and these concepts can be used to generate new images [23]. They measured the accuracy and diversity where high accuracy means that a model understands the meaning of symbol and high diversity means the samples have the variety in terms of the unspecified attributes. The SCAN model showed good performance on accuracy and diversity. Even if these models were effective in learning disentangled representations, they required the paired training data with labels and could not overcome the instability problem of GAN objective.

Meanwhile, there exists a method that learns disentangled representations with unsupervised learning. Unsupervised learning is more desirable because not only the labeling cost is high but also the labels annotated by human may be inconsistent and insufficient [24] Kim and Mnih proposed an unsupervised factor VAE, which enhanced disentanglement over β-VAE by introducing the total correlation penalty [24]. However, they conducted the experiment only for the artificial dataset. The hossRBM successfully learned disentangled representation in an unsupervised way on Toronto Face Dataset, which showed emotional changes in generated images. However, hossRBM can only learn discrete latent factors, and the computation cost grows exponentially. InfoGAN model made it possible to learn interpretable representation in purely unsupervised way by introducing the mutual information concept of latent code and input representation [25]. InfoGAN can learn both discrete and continuous latent factors as interpretable representations. Also InfoGAN usually requires the same amount of training time of typical GANs. Although this model learned interpretable and meaningful representations, it still had convergence or modal collapse problem because of binary classification discriminator. Therefore, we aim at developing a generative model that learns interpretable distributed representations and improves training GAN model.

In this study, we proposed IBEGAN model which introduces mutual information regularization between latent codes of the generator and latent representations of the autoencoder model of the discriminator. In our model, the training procedure follows the one of the BEGAN model for stable training and high visual quality.

In the remainder of the paper, we first review the related GAN models such as BEGAN and InfoGAN in Section 2. In Section 3, we describe our proposed model, information-based boundary equilibrium generative adversarial networks (IBEGANs), in terms of GAN objective functions and model architecture. In section 4, we evaluate whether our model can learn interpretable representations by manipulating the generated images through the latent codes in real-world image datasets, and the analyses of the image quality and the model convergence are performed. Finally, section 5 concludes this study.

2. Related Work

There are many GAN models to improve the generated input quality and obtain effective latent representations. Energy-Based GANs (EBGANs) [6] attempted to use an energy function to model the discriminator $D(\mathbf{x})$ based on the autoencoder model. Deep Convolutional GANs (DCGANs) [4] first used convolutional layer architecture and produced significantly improved visual samples. Wasserstein GANs (WGANs) [26] proposed to use Wasserstein distance as a measure of distance, which led to stable convergence. In this section, we introduce two GAN models that improve training procedure and learn interpretable representation: BEGAN and InfoGAN.

2.1. Boundary Equilibrium Generative Adversarial Networks. BEGAN construct GAN objective by using Wasserstein distance for autoencoder model. This method balances the generator and discriminator and also provides a new

approximate convergence measure. BEGAN has a simpler architecture and easier training procedure compared to other typical GANs. As a result, BEGAN can produce the samples of human faces, with the best visual quality.

BEGAN has proposed to match the loss distributions of autoencoders instead of matching data distributions directly. BEGAN uses the loss for a pixel wise autoencoder $\mathscr{L}$: $\mathbb{R}^{N_x} \longrightarrow \mathbb{R}^{N_x}$ as:

$$\mathscr{L}(\mathbf{x}) = |\mathbf{x} - D(\mathbf{x})|^{\eta} \quad \text{where} \begin{cases} D: \ \mathbb{R}^{N_x} \longrightarrow \mathbb{R}^{N_x} & \text{is the auto} - \text{encoder function,} \\ \mathbf{x} \in \mathbb{R}^{N_x} & \text{is a sample of dimension } N^x, \\ \eta \in \{1, 2\} & \text{is the target norm.} \end{cases} \tag{1}$$

Let ν_1, ν_2 be loss distributions of autoencoders, $C(\nu_1, \nu_2)$ be the set of couplings of ν_1 and ν_2, and $m_1, m_2 \in \mathbb{R}$ be the respective means. Then, Wasserstein distance is bounded by Jensen's inequality as follows in [3]:

$$W(\nu_1, \nu_2) = \inf_{\gamma \in C(\nu_1, \nu_2)} \mathbb{E}_{(\mu_1, \mu_2) \sim \gamma} \left[\left| \mu_1 - \mu_2 \right| \right] \geq \left| m_1 - m_2 \right|, \tag{2}$$

where μ_1, μ_2 have the same marginal distributions with ν_1, ν_2.

From this inequality, GAN loss is designed to maximize the lower bound of Wasserstein distance between autoencoder losses. Let θ_{D} and θ_{G} be the parameters of the discriminator and the generator, respectively, $\mathscr{L}_{\mathrm{D}}$ and $\mathscr{L}_{\mathrm{G}}$ be the losses of the discriminator and the generator, respectively, and $\mathbf{z}$ be the latent representation from autoencoder model. Then, the GAN objective can be expressed as follows:

$$\begin{cases} \mathscr{L}_{\mathrm{D}} = \mathscr{L}(\mathbf{x}) - \mathscr{L}(G(z_{\mathrm{D}}; \theta_{\mathrm{G}})), & \text{for } \theta_{\mathrm{D}}, \\ \mathscr{L}_{\mathrm{G}} = -\mathscr{L}_{\mathrm{D}}, & \text{for } \theta_{\mathrm{G}}. \end{cases} \tag{3}$$

In the GAN model, balancing losses of the discriminator and the generator is difficult because the discriminator usually wins over the generator easily. Therefore, BEGAN introduces a new hyperparameter $\gamma \in [0, 1]$ for stable convergence as well as balancing goal as follows:

$$\gamma = \frac{\mathbb{E}[\mathscr{L}(G(\mathbf{z}))]}{\mathbb{E}[\mathscr{L}(\mathbf{x})]}. \tag{4}$$

If γ becomes lower, the discriminator will focus on learning how to autoencode the real images. So image diversity will decrease. So BEGAN called γ as the diversity ratio.

BEGAN proposed to use proportional control theory to keep the equilibrium balanced at $\gamma \mathbb{E}[\mathscr{L}(\mathbf{x})] = \mathbb{E}[\mathscr{L}(G(\mathbf{z}))]$ by using a new variable $k_i \in [0, 1]$ which controls how much to focus on the loss of the generator during gradient descent. Finally, the full BEGAN objective is as follows:

$$\begin{cases} \mathscr{L}_{\mathrm{D}} = \mathscr{L}(\mathbf{x}) - k_i \mathscr{L}(G(\mathbf{z}_{\mathrm{D}})), & \text{for } \theta_{\mathrm{D}}, \\ \mathscr{L}_{\mathrm{G}} = \mathscr{L}(G(\mathbf{z}_{\mathrm{G}})), & \text{for } \theta_{\mathrm{G}}, \\ k_{i+1} = k_i + \lambda_{\mathrm{k}}(\gamma \mathscr{L}(\mathbf{x}) - \mathscr{L}(G(\mathbf{z}_{\mathrm{G}}))), & \text{for each training step } i, \end{cases} \tag{5}$$

where λ_k is the learning rate for k.

BEGAN has stable convergence and simple training procedure because it is not required to pre-train the discriminator and to train the discriminator and the generator alternatively. Nevertheless, this model cannot obtain interpretable representation which is useful to generate new samples from the generative model. Therefore, in the following section, we explain the InfoGAN model which learns the meaningful latent code.

2.2. Information Maximizing Generative Adversarial Networks. InfoGAN introduced the information theoretical modification to original GAN. It enabled the model to learn interpretable representations, which inspired our proposed method. InfoGAN relates the latent variable to the input variable by maximizing the lower bound of mutual information.

InfoGAN decomposes the input noise vector to the incompressible noise $\mathbf{z}$ and structured semantic latent code $\mathbf{c}$. The latent code $\mathbf{c}$ is a concatenation of latent variables $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N$ that were assumed to have factored distribution $P(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N) = \prod_{i=1}^{N} P(\mathbf{c}_i)$. In InfoGAN model, the incompressible noise $\mathbf{z}$ and the latent code $\mathbf{c}$ are both fed to the generator network, so the form of the generator becomes $G(\mathbf{z}, \mathbf{c})$, but the generator can ignore the information that $\mathbf{c}$ contains. To solve this problem, the mutual information of $\mathbf{c}$ and $G(\mathbf{z}, \mathbf{c})$ is maximized to restrict the generator on using the noise in a highly entangled way, where the lower bound of mutual information $I(\mathbf{c}; G(\mathbf{z}, \mathbf{c}))$ can be derived by defining an auxiliary distribution $Q(\mathbf{c}|\mathbf{x})$.

Hence the minimax game of InfoGAN is defined as follows:

$$\min_{\mathrm{G,Q}} \max_{D} \mathscr{L}_{\mathrm{InfoGAN}}(G, D, Q) = \mathscr{L}(G, D) - \lambda L_{\mathrm{I}}(G, Q), \tag{6}$$

where $L_I(G, Q) = H(\mathbf{c}) + \mathbb{E}_{\mathbf{c} \sim P(\mathbf{c}), \mathbf{x} \sim G(\mathbf{z}, \mathbf{c})}[\log Q(\mathbf{c}|\mathbf{x})]$.

InfoGAN successfully captured disentangled representations in MNIST dataset, 3D rendered image dataset, and the SVHN dataset. Those tasks were all unsupervised tasks, and the interpretable representations learnt by InfoGAN were competitive with several representations learnt by existing supervised methods. However, InGAN could not be applied to high-quality image examples because the binary classification discriminator was used instead of pixelwise error. Therefore, we propose a new information-based GAN model effectively reflecting pixelwise error.

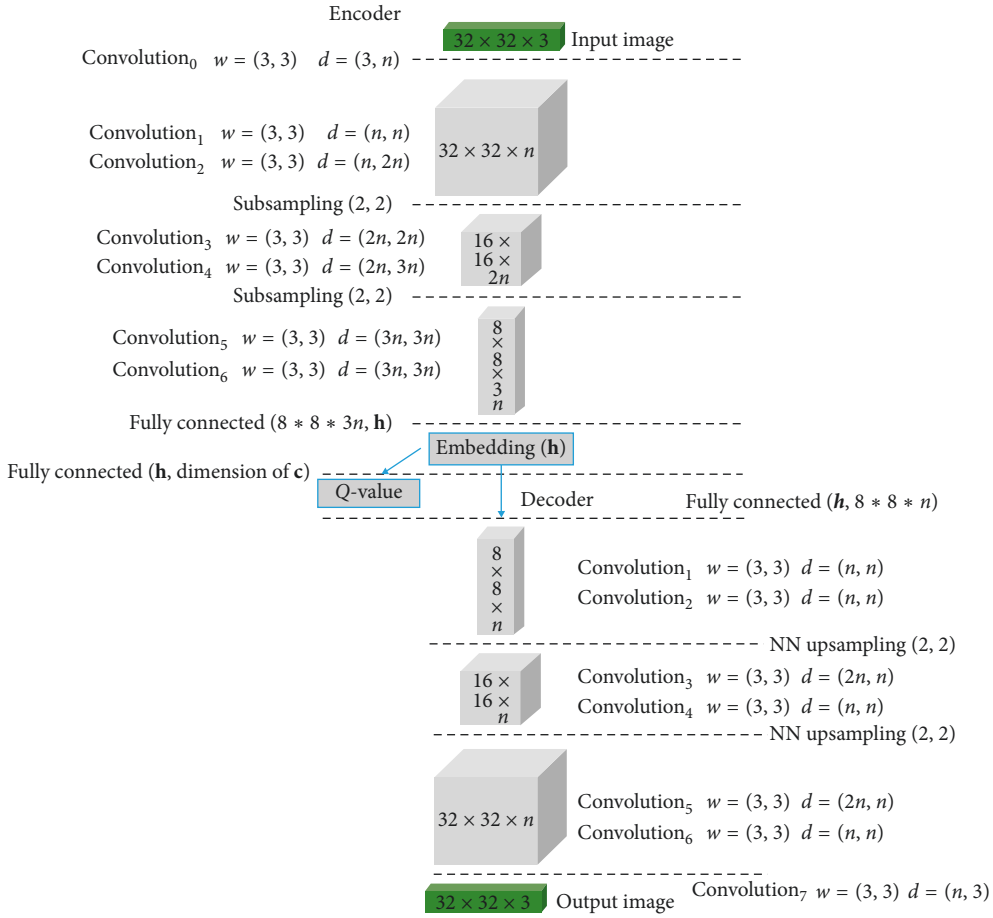


Figure 1: Network architecture of the discriminator.

3. Proposed Method

In this study, we proposed the IBEGAN model to generate and control new images with high visual quality and diversity. We used an autoencoder model with pixelwise error and proportional control theory to construct the discriminator loss as BEGAN model to alleviate convergence and diversity-quality balance problems. However, in the autoencoder model, we cannot restrict the kinds of latent representation such as discrete or continuous variable. IBEGAN consists of both continuous latent variable and discrete latent code. Therefore, we construct a new model architecture to learn interpretable representations with the discriminator using the autoencoder model.

First, our generator has two latent representations: incompressible noise $\mathbf{z}$ and latent code $\mathbf{c}$, which are used for the inputs of the generator $G(\mathbf{z}, \mathbf{c})$, where $\mathbf{z}$ learns compact representation of image and $\mathbf{c}$ learns an interpretable factor that can manipulate generating images. An autoencoder model with pixelwise loss is constructed, where autoencoder loss distributions is used for constructing the losses of the discriminator and generator. Unlike the InfoGAN model, IBEGAN can obtain a reproducible encoding $\mathbf{h}$ because an image can be obtained by decoding $\mathbf{h}$ in our model. However, like the InfoGAN model, the posterior distribution should be approximated by $Q(\mathbf{c}|\mathbf{x})$ because we cannot explicitly estimate a posterior $P(\mathbf{c}|\mathbf{x})$. If a new network architecture is used for $Q(\mathbf{c}|\mathbf{x})$, computational cost will be increased, so we share the encoder network to construct the network of $Q(\mathbf{c}|\mathbf{x})$ where $\mathbf{q}$ can be obtained from $\mathbf{h}$. In addition, we used a convolutional architecture to improve image quality. Figures 1 and 2 show the architecture of IBEGAN. The discriminator D: $\mathscr{R}^{N_x} \longrightarrow \mathscr{R}^{N_x}$ is a convolutional deep neural network [27, 28], designed as a deep autoencoder.

The generator G: $\mathscr{R}^{N_x} \longrightarrow \mathscr{R}^{N_x}$ has the same convolutional structure with the discriminator's decoder except the initial inputs and the first fully connected layer. This causes the simplicity of architecture, and the training procedure becomes simpler. The input of the generator is the concatenation of $\mathbf{z}$ and $\mathbf{c}$, where $\mathbf{z} \in [-1, 1]^{N_z}$ is sampled uniformly.

As a result, we construct the following loss functions from the networks:

$$\begin{cases} \mathscr{L}_{\mathrm{Q}} = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})\mathbf{c}\sim p(\mathbf{c}),\mathbf{x}\sim G(\mathbf{z},\mathbf{c})}[\log Q(\mathbf{c}|\mathbf{x})], \\ \mathscr{L}_{\mathrm{D}} = \mathscr{L}(\mathbf{x}) - k_i \mathscr{L}(G(\mathbf{z}, \mathbf{c})), \\ \mathscr{L}_{\mathrm{G}} = \mathscr{L}(G(\mathbf{z}, \mathbf{c})) - \lambda \mathscr{L}_{\mathrm{Q}}, \\ k_{i+1} = k_i + \lambda_{\mathrm{k}}(\gamma \mathscr{L}(\mathbf{x}) - \mathscr{L}(G(\mathbf{z}, \mathbf{c}))), \end{cases} \tag{7}$$

where $\mathscr{L}(\mathbf{x})$ is an autoencoder loss as in Equation (3).

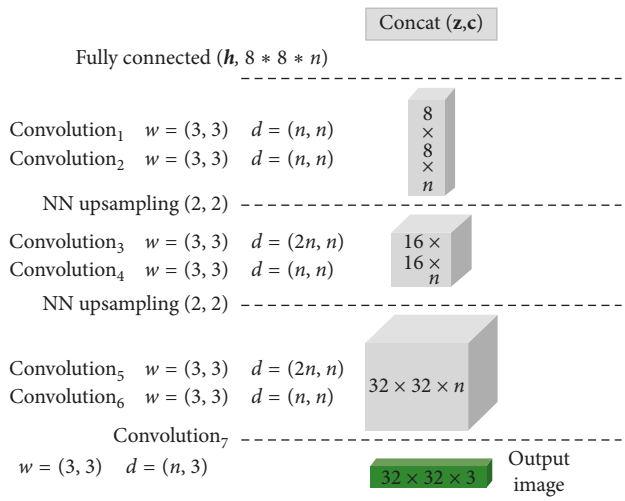


Figure 2: Network architecture of the generator.

Figure 3: Samples of CelebA dataset.

We trained an autoencoder network for the discriminator loss $\mathscr{L}_D$, a generator network for the generator loss $\mathscr{L}_G$, where $\mathscr{L}_Q$ regularizes the generator not to ignore latent code **c**. In our objectives, the meaning of γ can be preserved as in the BEGAN model because we update k_i regardless of the regularizer $\mathscr{L}_Q$. Therefore, we will measure the convergence of GANs as in [3].

4. Experiments

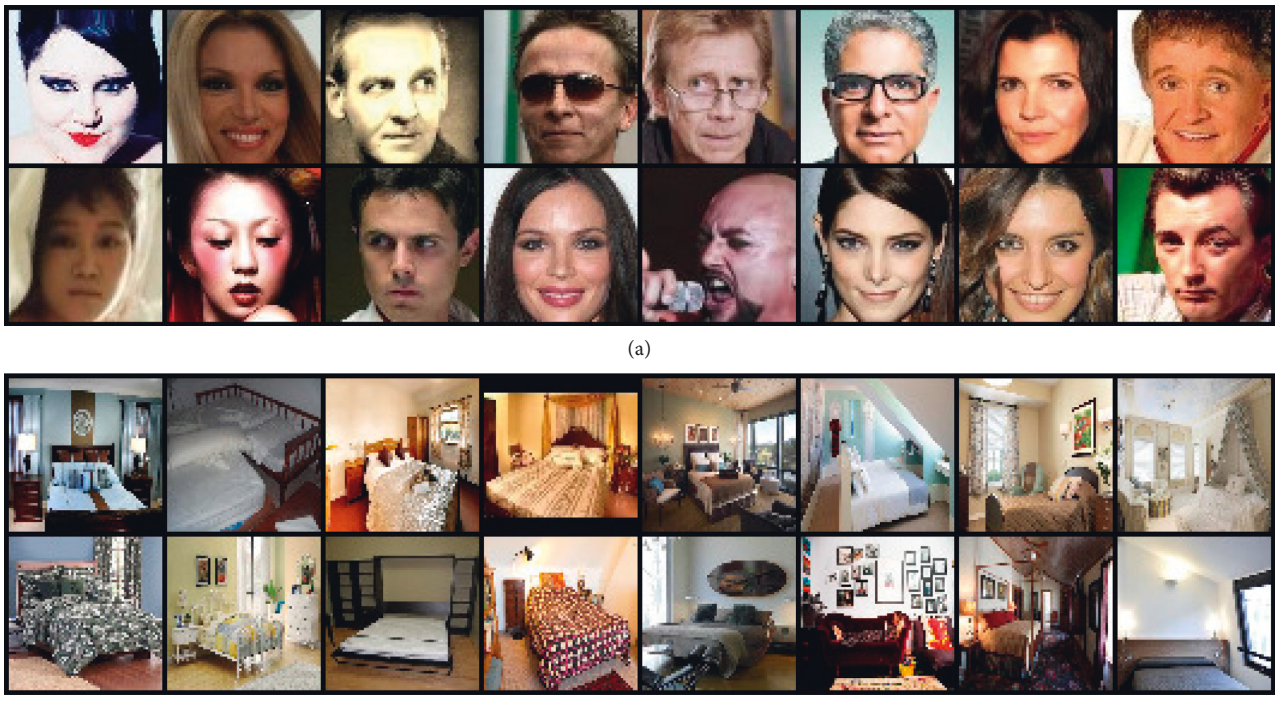
The goal of experiments is to evaluate if our proposed method can learn interpretable representations from image datasets while assuring the newly produced images to maintain high visual quality. Therefore, our experiment consists of two parts. First, we show the generated images while changing the learned latent code **c** and fixing the latent variable **z** and interpret the results. To evaluate the learned interpretable codes, we measure the accuracy of the generated samples. Second, we measure whether IBEGAN converges to the equilibrium by inspecting the change of loss and using convergence measure.

4.1. Data Description. In this paper, we used two real-world image datasets: CelebA and LSUN bedroom.

Figure 3 shows CelebA dataset, which contains 202,599 images of celebrities. CelebA has a various facial poses, various races, and rotations around the camera axis [29]. Our proposed model generated the images that seem realistic and captured meaningful and manipulative representations.

Figure 4 shows LSUN bedrooms dataset, which contains about 3,033,042 images of bedrooms [30]. Though LSUN is a high resolution dataset with a lot of samples, the proposed

Figure 4: Samples of LSUN bedroom dataset.



(a)

(b)

Figure 5: Resized face data and bedroom data that we used in this experiment.

model can successfully learn the representations and generate high-quality samples.

In our experiments, we resized the image size of both CelebA data and LSUN bedroom data to 64×64, and with CelebA data, we used center crop of the image. The face data that were used in training are illustrated in Figure 5.

4.2. Experimental Design. In the experiments, we construct the network architectures of the discriminator and the generator as in Figures 1 and 2, where 3×3 convolutions with ELU (exponential linear units) layers are repeatedly applied at the previous outputs [31]. Each layer is repeated twice because the repetition of convolution can produce

Figure 6: Varying $\mathbf{c}_1$ on CelebA dataset. We show the effect of the learned categorical latent factors on the outputs. This figure shows that a categorical code $\mathbf{c}_1$ can capture the gender of the face. We can see that the degree of masculinity of the image changes drastically as latent factor $\mathbf{c}_1$ changes.

better visual quality. In encoder network, downsampling is done by subsampling with 2 strides. In decoder and generator networks, upsampling is done by using the nearest neighbor method. At the end of the encoder, the intermediate input is passed to a fully connected layer, and the embedding $\mathbf{h}$, new input of the decoder is obtained. To capture the auxiliary distribution Q, the embedding $\mathbf{h}$ is passed to the next fully connected layer with the same output dimensions as the sum of the dimensions of the latent codes.

We estimated an auxiliary distribution $Q(\mathbf{c}|\mathbf{x})$ by using Q network which shares a lot of parts with the encoder network and constructed $\mathscr{L}_Q$ from the softmax cross entropy between $\mathbf{c}$ and the estimated Q.

$$\mathscr{L}_Q = \frac{1}{M} \sum_{i=1,\ldots,N_c} \sum_{j=1,\ldots,m_i} \left(\mathbf{c}_{i,j} * \mathbf{q}_{i,j}\right), \tag{8}$$

where $N_{\mathbf{c}}$ is the number of latent codes, m_i is the dimension of the latent code i, $M = \sum_i m_i$, $\mathbf{c}_{i,j}$, is j-th category of the latent code i, and $\mathbf{q}_{i,j} = \log Q(\mathbf{c}_i|\mathbf{x})_j$. This regularizer can make IBEGAN learn the meaningful latent codes. We will inspect this property in Section 4.3.1.

In [23], the effectiveness of disentangling factors was proved by measuring the accuracy, but IBEGAN cannot be verified in that way because the interpretable factors are learned in a unsupervised way. Although IBEGAN does not have any supervised label for the latent code unlike [23], the latent code should be determined to generate samples. Therefore, we used the latent code as the supervised label and estimated the most probable latent code from the auxiliary distribution. We generate artificial images $\tilde{\mathbf{x}}$ by changing the latent codes $\tilde{\mathbf{c}}$, estimating $Q(\mathbf{c}|\tilde{\mathbf{x}})$, and comparing the estimated codes $\hat{\mathbf{c}} = \text{argmax}_{\mathbf{c}}\, Q(\mathbf{c}|\tilde{\mathbf{x}})$ with the latent codes $\tilde{\mathbf{c}}$. We calculate the accuracy of the estimation to provide the objective measure for the interpretability of our generated images.

To train the model, all two losses $\mathscr{L}_{\mathrm{D}}$ and $\mathscr{L}_{\mathrm{G}}$ in Equation (7) are minimized in an iteration, and the variable

Figure 7: Varying $\mathbf{c}_2$ on CelebA dataset. This figure shows that a categorical code $\mathbf{c}_2$ can capture the hair style. As $\mathbf{c}_2$ changes, we can find that hair style changes while a person's face is same.

$k_i \in [0, 1]$ is updated to control how much to concentrate on the loss of the generator versus the loss of the data, $\mathbf{x}$, during the gradient descent. We used the different basic settings of some hyperparameters for CelebA and LSUN bedroom datasets because images in CelebA datasets share more common features than LSUN. In both basic settings, we used a small batch size $m = 16$ to avoid a memory error, and the Adam optimizer was used to minimize the losses [32]. In addition, diversity ratio γ was set to 0.5 and fixed. However, we set the dimension of hidden representation as 64 for CelebA and 128 for LSUN bedroom. In case of a learning rate of k λ_k, we used 0.001 for CelebA dataset and 0.0005 for LSUN bedroom dataset.

We should carefully select a control parameter λ of the regularization term $\mathscr{L}_Q$ in Equation (7). Unlike the InfoGAN model, we could not set λ to 1 because we used an autoencoder-based discriminator loss instead of a classifier-based loss. We initialized λ to $1/M^2$ and updated with the learning rates of the generator and the discriminator. However, unlike the learning rates, λ was increased when it was updated.

To measure the convergence of the model, we used a convergence measure that is proposed in [3].

$$\mathscr{M}_{\text{conv}} = \mathscr{L}(\mathbf{x}) + \left|\gamma \mathscr{L}(\mathbf{x}) - \mathscr{L}(G(\mathbf{z}, \mathbf{c}))\right|. \tag{9}$$

Because IBEGAN uses the proportion control algorithm, $\mathscr{M}_{\text{conv}}$ is stabilized when the model reaches a final stage. If

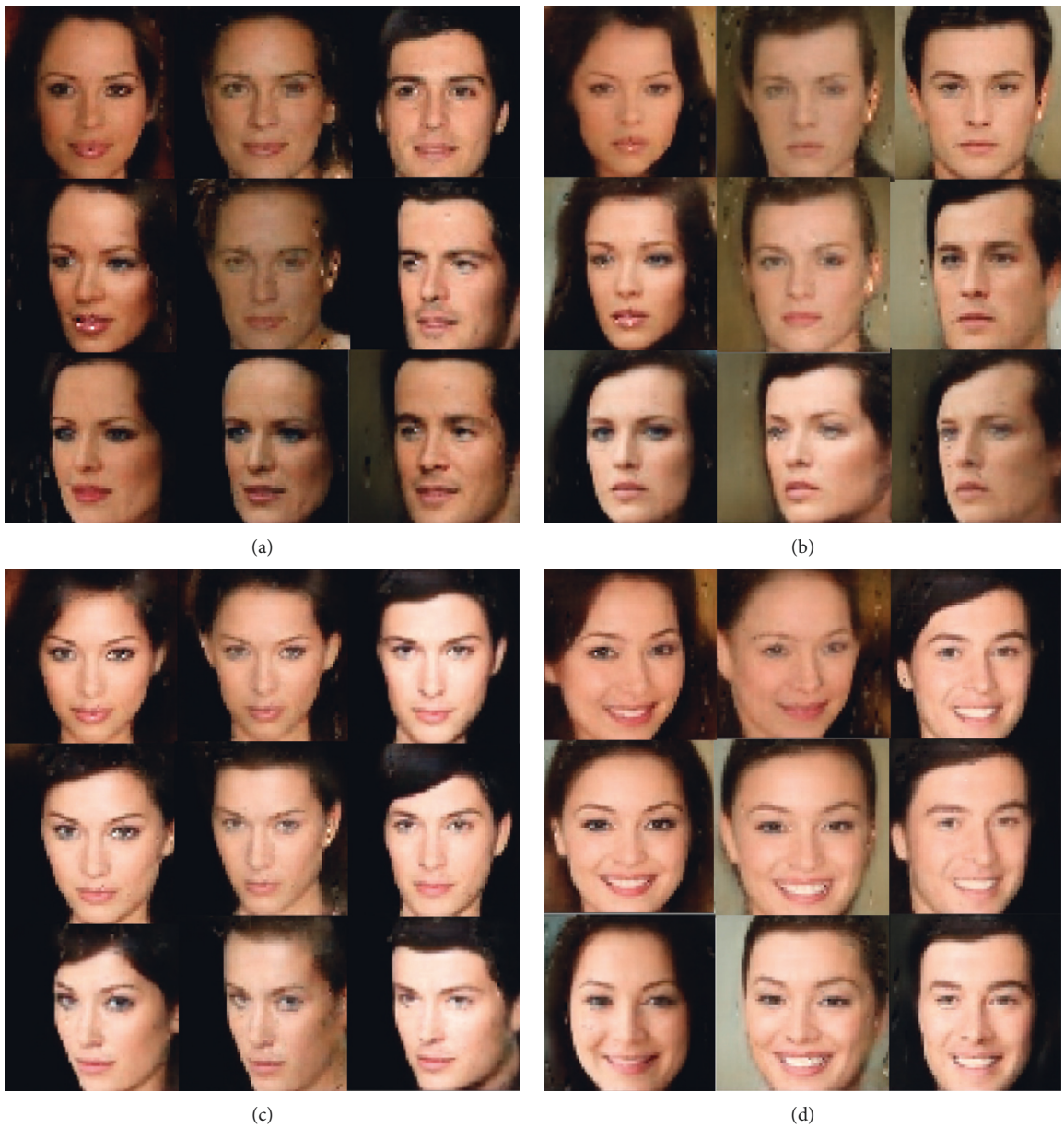


FIGURE 8: Variation of both $\mathbf{c}_1$ and $\mathbf{c}_2$ with same incompressible noise $\mathbf{z}$. Adjoined nine images have same $\mathbf{z}$ with different latent codes $\mathbf{c}_1$ and $\mathbf{c}_2$. Row of the image denotes difference of $\mathbf{c}_1$ while column refers to $\mathbf{c}_2$.

a model collapses, $\mathscr{M}_{\text{conv}}$ cannot be stabilized. We analyze the convergence of our model from various loss values and this convergence measure in Section 4.3.2.

4.3. Experimental Results

4.3.1. Interpretable Representation. In this section, we evaluate our generative model in terms of interpretable representation. From the learned latent codes, we can find meaningful features and tendencies in spite of the absence of any additional label information. In addition, we quantitatively verify that our model can generate the manipulated images by changing the interpretable latent code $\mathbf{c}$.

To generate the artificial face images like CelebA dataset, first we choose to model the latent codes with two categorical codes, $\mathbf{c}_1 \sim \text{Cat}(K = 9, p = 1/9)$ and $\mathbf{c}_2 \sim \text{Cat}(K = 8, p = 1/8)$. In this setting, our suggested model learns to represent latent factors as gender and hair style. Figure 6 demonstrates that the variation of $\mathbf{c}_1$ is related to gender. Figure 7 shows that the variation of $\mathbf{c}_2$ changes the hair style of the generated images. We can control the generated image of a person to look more manly by changing $\mathbf{c}_1$, and we can try different hair styles on images by controlling $\mathbf{c}_2$, which proves that our proposed model successfully learned interpretable representations.

Aforementioned experimental results showed the latent factors of hair style and masculinity. As a result, two factors

Figure 9: Varying c on LSUN dataset. The size of bed changes drastically.

are highly correlated because men and women usually have different hair styles. We also experimented with different level of categorical code in CelebA dataset. In this experiment, we selected latent codes with two categorical codes, $\mathbf{c}_1 \sim \mathrm{Cat}(K = 10, p = 1/10)$ and $\mathbf{c}_2 \sim \mathrm{Cat}(K = 10, p = 1/10)$. In this setting, both latent factors indicated independent feature of the image.

Figure 8 shows the second experimental result in CelebA dataset. This image illustrates variation of both $\mathbf{c}_1$ and $\mathbf{c}_2$ with same incompressible noise $\mathbf{z}$. Nine adjoined images are from same incompressible noise $\mathbf{z}$ and row denote varying $\mathbf{c}_1$ and column refers to $\mathbf{c}_2$.

We can find that $\mathbf{c}_1$ can capture the angle of the face, since generated faces tend to look at the left side as $\mathbf{c}_1$ changes. Also, $\mathbf{c}_2$ refers to masculinity since generated image's gender differs as $\mathbf{c}_2$ changes. Moreover, we found that image with same $\mathbf{z}$ has common in its background, skin color, and facial expressions such as smile. We can conclude that incompressible noise $\mathbf{z}$ denotes overall air of the image.

Two experiments in CelebA showed that interpretable representations in various latent codes $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_k$, can be either independent or dependent because our model learns the latent codes without determining the desirable factors. In our first experiments in CelebA, two latent codes were correlated because the first code captured masculinity and the second code denoted hair style. In the second experiment settings, two latent codes captured hair style and angle of the face. Two latent codes in the second experiments were independent.

Table 1: The accuracy of the estimation of the interpretable codes based on $Q(\mathbf{c}|\mathbf{x})$.

Latent factor	Accuracy	Standard deviation
$\mathbf{c}_1$	0.9865	0.0051
$\mathbf{c}_2$	0.9930	0.0048
$\mathbf{c}_1$ and $\mathbf{c}_2$	0.9796	0.0080

We also implemented our suggested model to LSUN bedroom dataset, and the result is illustrated in Figure 9. In this experiment, we used latent codes with single categorical code with ten dimensions, $\mathbf{c}_1 \sim \mathrm{Cat}(K = 10, p = 1/10)$. We generated the images by fixing 10 different incompressible noises $\mathbf{z}$ and changing the one-hot encoding of the latent code $\mathbf{c}_1$. Therefore, in Figure 9, rows refer the same $\mathbf{z}$ and

(a)

(b)

Figure 10: Comparison of our suggested model IBEGAN with previous model InfoGAN. Both models captured variation of hair style by changing latent code $\mathbf{c}_1$. (a) Generated images in InfoGAN. (b) Generated images in IBEGAN (Suggested model).

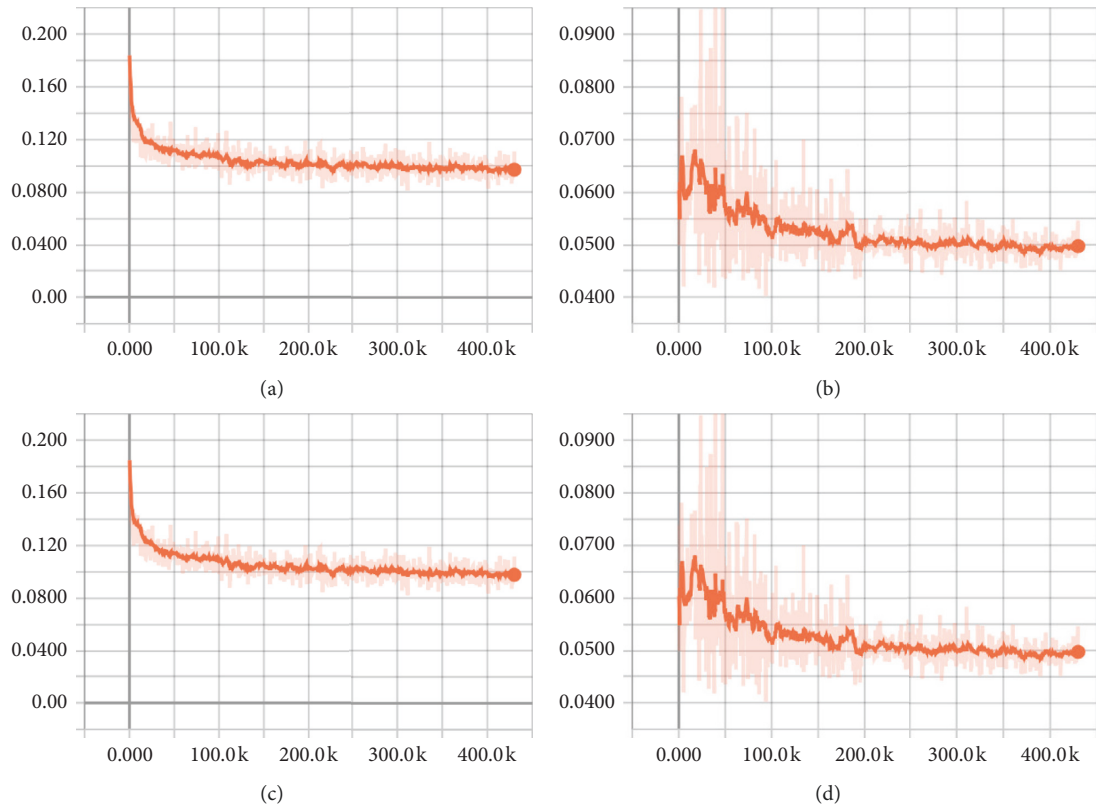


Figure 11: Various losses. (a) Discriminator loss. (b) Generator loss. (c) Autoencoder loss for real data. (d) Autoencoder loss for fake data.

columns refer the same $\mathbf{c}_1$. In this experiment, $\mathbf{c}_1$ captures size of bed in each image. We can see in Figure 9 that size of bed changes as latent code $\mathbf{c}_1$ changes. In addition, similarly to CelebA, the incompressible noise $\mathbf{z}$ can capture overall air of the image.

In this section, we also verify the effectiveness of the interpretable latent codes. We generate the new images and calculate the auxiliary distribution $Q(\mathbf{c}|\mathbf{x})$. To generate an image using generator network, feeding the latent codes is required. Therefore, we can naturally obtain the label information of the generated image. Table 1 illustrates the classification results based on auxiliary distribution $Q(\mathbf{c}|\mathbf{x})$. We repeated the experiments 100 times and provided the average and the standard deviation of the accuracy.

In Table 1, the first and the second rows illustrate the accuracy of the individual factor whereas the last row illustrates the accuracy of two factors at the same time. As expected, our learned interpretable codes are well reflected in the generated images and showed high accuracies and small standard deviations in all cases.

4.3.2. Convergence of Models. In this section, we demonstrate that the IBEGAN model can produce high visual quality and be stably trained. These properties come from the advantage of the BEGAN model.

We aim not only to learn interpretable representations but also to obtain high-quality images. To verify this, we compared generated images of CelebA dataset from ours with images from the InfoGAN model. This result is illustrated in Figure 10. Both generated images changes hair styles by changing latent code $\mathbf{c}_1$. We can see that our suggested model shows better image quality compared to the previous model.

To inspect the convergence of our model, we illustrate all loss values in Figure 11. We constructed the generator and discriminator losses based on autoencoder losses of real and fake data. Therefore, we drew the discriminator loss in Figure 11(a), the generator loss in Figure 11(b), the autoencoder loss of real data in Figure 11(c), and the autoencoder loss of fake data in Figure 11(d).

From the plots, we found that the pattern of the discriminator loss is similar to the autoencoder loss of real data and the pattern of the generator loss is similar to the autoencoder loss of fake data. It is because we set small λ, k_0, and λ_k. While the discriminator loss was stably declining, the generator loss was unstable in the beginning of the training but eventually stabilized. In this experiment, we set the balance parameter γ to 0.5. When the model converges, γ satisfies the balancing equation (4), where γ is the ratio of $\mathscr{L}(G(\mathbf{z}, \mathbf{c}))$ to $\mathscr{L}(\mathbf{x})$. At the end of the iteration, the autoencoder loss of real data is about twice the autoencoder loss of fake data.

In addition, we evaluate the convergence of our model with the convergence measure (9). Figure 12 shows the change of generated images with $\mathscr{M}_{\text{conv}}$. The fidelity of image is low when $\mathscr{M}_{\text{conv}}$ is high. Also, when $\mathscr{M}_{\text{conv}}$

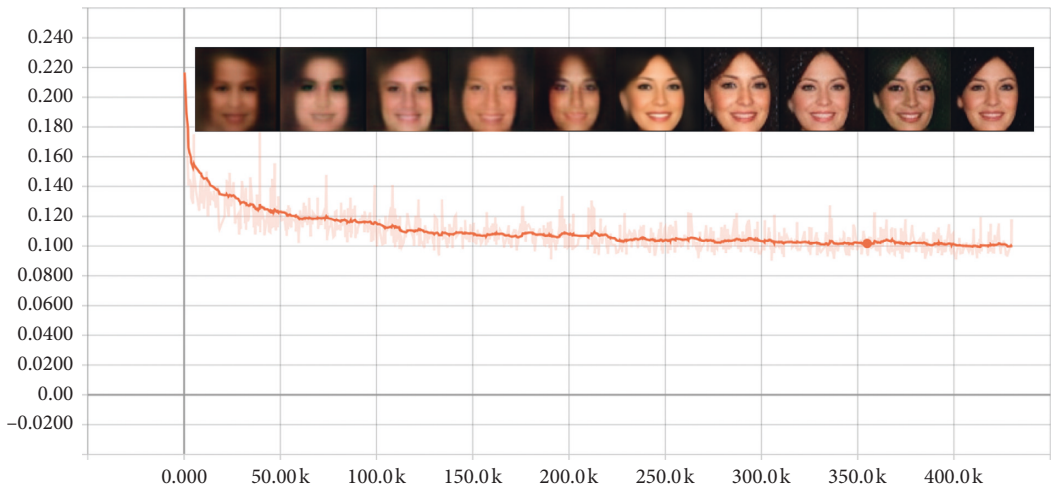


Figure 12: The convergence measure $\mathscr{M}_{\text{conv}}$.

become stabilized, the images hardly change. We find that the IBEGAN model stably converges.

5. Conclusion

This paper proposed a meta-algorithm called IBEGAN. IBEGAN does not require any kind of supervision and is still able to learn interpretable representations. In addition, the IBEGAN model can have a simple training procedure and generate high-quality images by introducing equilibrium concept and proportional control theory as in the BEGAN model. In the experiment, we can obtain produced images that maintain the high visual quality, and the images can be manipulated as desired by changing the values of latent codes.

In this study, we used an autoencoder model to sophisticate the discriminator, but we should additionally construct the generator network unlike the BEGAN model because the deterministic autoencoder model cannot control the distribution of hidden representation. Therefore, the IBEGAN model can be extended to sharing generator network with the encoder network by using variational autoencoder.

However, the IBEGAN model had a difficulty in learning latent codes independently. This model can be extended to adding a regularization term that could make latent factors independently. In this way, IBEGAN model would produce more robust disentangling latent codes that are independent.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) (No. 2016R1A2B3014030). Also, it was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIP) (No. 2017R1A5A1015626).

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," in *Proceedings of Advances in Neural Information Processing Systems*, pp. 2672–2680, Montréal, QC, Canada, December 2014.

[2] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *DeepLearning*, Vol. 1, MIT Press, Cambridge, MA,USA, 2016.

[3] D. Berthelot, T. Schumm, and L. Metz, "Began: boundary equilibrium generative adversarial networks," 2017, https://arxiv.org/abs/1703.10717.

[4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, https://arxiv.org/abs/1511.06434.

[5] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Proceedings of Advances in Neural Information Processing Systems*, pp. 2234–2242, Barcelona, Spain, June 2016.

[6] J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," 2016, https://arxiv.org/abs/1609.03126.

[7] Z. Dai, A. Almahairi, P. Bachman, E. Hovy, and A. Courville, "Calibrating energy-based generative adversarial networks," 2017, https://arxiv.org/abs/1702.01691.

[8] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[9] Y. Bengio, "Learning deep architectures for ai," *Foundations and Trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[10] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[11] G. E. Hinton, P. Dayan, B. J. Frey, and R. M. Neal, "The "wake-sleep" algorithm for unsupervised neural networks," *Science*, vol. 268, no. 5214, pp. 1158–1161, 1995.

[12] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Proceedings of Advances in Neural Information Processing Systems*, pp. 3581–3589, Montreal, QC, Canada, December 2014.

[13] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun, "Disentangling factors of variation in deep representation using adversarial training," in *Proceedings of Advances in Neural Information Processing Systems*, pp. 5040–5048, Barcelona, Spain, December 2016.

[14] A. Odena, "Semi-supervised learning with generative adversarial networks," 2016, https://arxiv.org/abs/1606.01583.

[15] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," 2015, https://arxiv.org/abs/1511.06390.

[16] C. Wang, C. Wang, C. Xu, and D. Tao, "Tag disentangled generative adversarial networks for object image re-rendering," in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, Melbourne, VIC, Australia, August 2017.

[17] T. Xiao, J. Hong, and J. Ma, "DNA-GAN: learning disentangled representations from multi-attribute images," 2017, https://arxiv.org/abs/1711.05415.

[18] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[19] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, pp. 1096–1103, Helsinki, Finland, July 2008.

[20] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther, "Improving semi-supervised learning with auxiliary deep generative models," in *Proceedings of NIPS Workshop on Advances in Approximate Bayesian Inference*, Montreal, QC, Canada, December 2015.

[21] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural Computation*, vol. 12, no. 6, pp. 1247–1283, 2000.

[22] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," 2015, https://arxiv.org/abs/1511.05644.

[23] I. Higgins, N. Sonnerat, L. Matthey et al., "SCAN: learning hierarchical compositional visual concepts," in *Proceedings of International Conference on Learning Representations*, Sydney, NSW, Australia, December 2018.

[24] H. Kim and A. Mnih, "Disentangling by factorising," 2018, https://arxiv.org/abs/1802.05983.

[25] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: interpretable representation learning by information maximizing generative adversarial nets," in *Proceedings of Advances in Neural Information Processing Systems*, pp. 2172–2180, Barcelona, Spain, December 2016.

[26] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017, https://arxiv.org/abs/1701.07875.

[27] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: a review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[28] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.

[29] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, Las Condes, Chile, December 2015.

[30] F. Y. Y. Z. S. Song and A. S. J. Xiao, "Construction of a large-scale image dataset using deep learning with humans in the loop," 2015, https://arxiv.org/abs/1506.03365.

[31] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units," 2015, https://arxiv.org/abs/1511.07289.

[32] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.