Improved ArtGAN for Conditional Synthesis of Natural Image and Artwork

Wei Ren Tan, *Student Member, IEEE*, Chee Seng Chan, *Senior Member, IEEE*, Hernán E. Aguirre, *Member, IEEE*, and Kiyoshi Tanaka, *Member, IEEE*

***Abstract*— This paper proposes a series of new approaches to improve generative adversarial network (GAN) for conditional image synthesis and we name the proposed model as "ArtGAN." One of the key innovation of ArtGAN is that, the gradient of the loss function w.r.t. the label (randomly assigned to each generated image) is back-propagated from the categorical discriminator to the generator. With the feedback from the label information, the generator is able to learn more efficiently and generate image with better quality. Inspired by recent works, an autoencoder is incorporated into the categorical discriminator for additional complementary information. Last but not least, we introduce a novel strategy to improve the image quality. In the experiments, we evaluate ArtGAN on CIFAR-10 and STL-10 via ablation studies. The empirical results showed that our proposed model outperforms the state-of-the-art results on CIFAR-10 in terms of Inception score. Qualitatively, we demonstrate that ArtGAN is able to generate plausible-looking images on Oxford-102 and CUB-200, as well as able to draw realistic artworks based on style, artist, and genre. The source code and models are available at: https://github.com/cs-chan/ArtGAN.**

***Index Terms*— Generative adversarial networks, deep learning, image synthesis, artwork synthesis, ArtGAN.**

I. Introduction

"Good artists copy, great artists steal." [65]
– Pablo Picasso

RECENTLY, Goodfellow *et al.* [1] proposed an interesting features learning model called Generative Adversarial Networks (GAN) by employing two neural networks that are adversarially trained. Unlike the traditional deep discriminative models [2]–[4], the representations learned by GAN can be visualized through the generator in GAN in the form of synthetic images. More interestingly, these generated images look more realistic to human observers compared to other generative models. Since then, many extensions of GAN [5]–[11] have been introduced and showed promising results in generating appealing images when trained on datasets, such as MNIST [12], CIFAR-10 [13], ImageNet [14], etc. Despite the success, there is still room for improvement as the synthetic image quality is still far from realistic.

While unconditional GAN is an important research area, this paper is interested in class-conditioned GAN. In particular, conditional GAN is useful to understand how the visual representation of each class is learned via the visualization techniques inherent in GAN. Furthermore, we are interested to investigate if a machine can generate artwork based on style, genre, or artist. Artwork is a mode of creative expression, coming in different kinds of forms, including drawing, naturalistic, abstraction, etc. Unlike the aforementioned datasets [12]–[14], the representations of artworks can be harder to learn because they are usually non-figurative or abstract.

To this end, we propose a novel conditional GAN named **ArtGAN** for conditional synthesis of natural image and artwork. We anticipate that a good way to look at this problem is to understand how humans learn to draw. An artist teacher wrote an online article[1] and pointed out that an effective learning requires to focus on a particular type of skills at a time, e.g. practice to draw a particular object or one kind of movement at a time. Accordingly, ArtGAN takes a randomly chosen label information and a noise vector as inputs. The chosen label is used as the true label when computing the loss function for the generated image. The idea is to allow the generator to learn more efficiently by leveraging the feedback information from the labels. Inspired by recent works [15], [16], a categorical autoencoder-based discriminator that incorporates an autoencoder into the categorical discriminator for additional complementary information is introduced. Rather than deploying two separate computationally expensive networks (i.e. a categorical discriminator and an autoencoder separately), the categorical autoencoder-based discriminator in our proposed GAN partly shares the same architecture and weights. In specific, *encoder* in the autoencoder is shared by the categorical discriminator as illustrated in Figure 1.

In addition, we introduce a novel strategy to improve the generated image quality. The motivation behind this strategy is to generate a set of pixels that vote for a better quality pixel via average ranking in order to generate better pixel

Manuscript received August 30, 2017; revised April 28, 2018 and July 14, 2018; accepted August 13, 2018. Date of publication August 22, 2018; date of current version September 25, 2018. This work was supported in part by the Fundamental Research Grant Scheme (FRGS) MoHE from the Ministry of Education Malaysia under Grant FP004-2016 and in part by the UM Frontier Research from University of Malaya under Grant FG002-17AFR. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Catarina Brites. *(Corresponding author: Chee Seng Chan.)*

W. R. Tan, H. E. Aguirre, and K. Tanaka are with Shinshu University, Nagano 380-8553, Japan (e-mail: 14st203c@shinshu-u.ac.jp; ahernan@shinshu-u.ac.jp; ktanaka@shinshu-u.ac.jp).

C. S. Chan is with the Center of Image and Signal Processing, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia (e-mail: cs.chan@um.edu.my).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2866698

[1] http://www.learning-to-see.co.uk/effective-practice

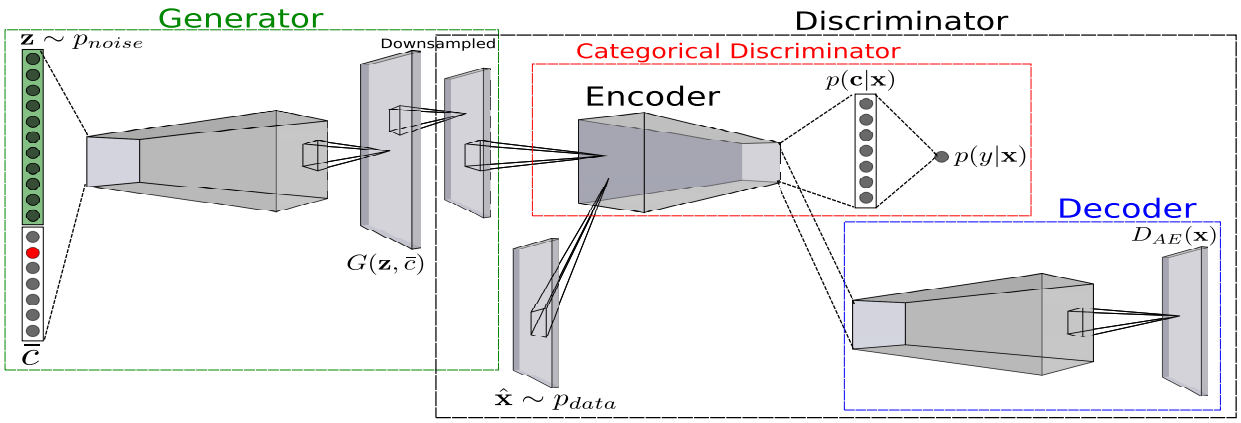


Fig. 1. Overview of ArtGAN-AEM architecture. $\mathbf{z}$ and $\hat{c}$ are concatenated and fed to the generator to produce synthetic image $G(\mathbf{z}, \hat{c})$. Either the downsampled generated image $G(\mathbf{z}, \hat{c})$ or real data $\hat{\mathbf{x}}$ is used as the input $\mathbf{x}$ to the (categorical autoencoder-based) discriminator. The discriminator produces three outputs: the class prediction $p(\mathbf{c}|\mathbf{x})$, adversarial prediction $p(y|\mathbf{x})$, and the reconstructed image $D_{AE}(\mathbf{x})$.

values. One may naively train an ensemble GANs to achieve this goal. However, training multiple GANs explicitly is computationally expensive and does not guarantee to achieve similar performance gain [17], [18]. Hence, we innovate an alternative approach where the generator in ArtGAN will generate synthetic images with resolution $2\times$ higher than the original image size. Then, these generated images will be downsampled to the original size using averaged pooling operation as a form of voting scheme. Example of the ArtGAN outputs are illustrated at Figure 2.

In summary, our key contributions are: i) We propose ArtGAN to emulate the concept of effective learning to generate very challenging images. Within this, we introduce a novel way to improve the image quality. ii) Empirically, we show that our proposed models are able to generate CIFAR-10 [13] and STL-10 [19] images with better Inception scores compared to the state-of-the-art results. iii) Our models are capable of generating Oxford-102 [20] and CUB-200 [21] samples that contain clear object structures in them. At the same time, ArtGAN is also able to generate high quality artwork that exhibit similar visual representations within *genre*, *artist*, or *style*. To the best of our knowledge, no existing empirical research has addressed the implementation of a generative model on a large scale artworks dataset.

A preliminary version of this work was presented earlier [22]. The present work adds to the initial version in significant ways. First, we extend ArtGAN with the introduction of categorical autoencoder-based discriminator. Secondly, we innovate a way to improve the image quality generated by ArtGAN. Thirdly, considerable new analysis and intuitive explanations are added to the initial results. For instance, we extend the original qualitative experiments from Wikiart [23] to CIFAR-10 [13], STL-10 [19], Oxford-102 [20], and CUB-200 [21] datasets. In addition, we included the Inception score [24] as a quantitative metric where ArtGAN obtained state-of-the-art result on CIFAR-10 dataset.

The rest of the paper is structured as follows. Related works are discussed in the next section (Section II). Section III describes the proposed models, while the image quality strategy is explained in detail in Section IV. Experiments are discussed in Section V. Last but not least, conclusion is drawn in Section VI.

II. Related Works

Generative models have been a fundamental interest and challenging problem in the field of computer vision and machine learning. In contrast to discriminative models which only allow sampling of the target variables conditioned on the observed quantities, generative models can be used to simulate observed distribution, and so they offer a much richer representation. Early works [25]–[27] studied the statistical properties of natural images, but are limited to texture or certain patterns (e.g. faces) only due to the difficulty in learning an effective feature representation. Recently, advances in deep models nourish a series of deep generative models [28], [29] for image generation through the Bayesian inference, typically trained by maximizing the log-likelihood. These models are able to construct decent quality images on less complicated images, such as digits and faces, but generally have intractable likelihood and require numerous approximations. Denoising autoencoders (DAE) [30] were introduced to overcome the intractable problem, but the reconstructed images are generally blurry. Then, DRAW [31] was proposed, depicted as a sequential model with attention mechanism to draw image recursively. It mimics the process of human drawing but faces challenges when it is scaled up to large and complex images. PixelRNN [32] is another autoregressive approach for image generation that has received much attentions recently. Its extensions (PixelCNN [33] and PixelCNN++ [34]) are able to synthesize decent images but are computationally expensive to train.[2]

Recently, a more significant breakthrough framework, Generative Adversarial Network (GAN) was introduced by Goodfellow *et al.* [1]. This framework escapes the difficulty of maximum likelihood estimation by estimating the generative model via an adversarial process and has gained striking successes in natural image generation. However, GAN is

[2]They reported that PixelCNN++ requires approximately 5 days to converge to the reported results using 8 Maxwell TITAN X GPUs in github: https://github.com/openai/pixel-cnn.

Fig. 2. ArtGAN samples on Wikiart, CIFAR-10, STL-10, Oxford-102 and CUB-200 (Best viewed in color).

well-known for its instability during training. To tackle this problem, feature matching [24] was proposed to generate descent quality images. Instance noise [35] is also an effective method to remedy the instability problem. Several variants proposed to address this problem by analysing the objective function of GAN. Wasserstein GAN (WGAN) used the Lipschitz constrained Earth-Mover (EM) distance to address the vanishing gradient and the saturated Jensen-Shannon distance problems. However, WGAN can still generate low quality images and fail to converge in many settings. An improvement [36] was proposed to overcome these problems. Although they argued that the performance is more stable at convergence, WGAN is still outperformed by DCGAN [6] in terms of convergent speed and Inception score. A similar solution was introduced in Loss-Sensitive GAN (LS-GAN) [37] with theoretical analysis on Lipschitz densities. They conceptually proved that the GAN loss functions with bounded Lipschitz constants are sufficient to match the model density to true data density. However, objects in their generated CIFAR-10 images are hardly recognizable. Meanwhile, Least Squares GAN (LSGAN) [38] adopted the least square loss function in the discriminator. They showed that minimizing the objective function yields minimizing the Pearson $\mathcal{X}^2$ divergence. Their results demonstrated that LSGAN is able to synthesize appealing images on LSUN, CIFAR-10, and handwritten Chinese characters datasets.

Presently, another subfamily of GAN was introduced where an autoencoder is employed in the discriminator. The Energy-based GAN (EBGAN) [15] is trained by replacing the discriminator with an autoencoder and it has demonstrated decent quality synthetic images up to 256×256 pixels. Denoising Feature Matching (DFM) [39] maintains the traditional GAN adversarial loss, but an additional complementary information to the generator is computed using a denoising autoencoder in the feature space learned by the discriminator. DFM achieved state-of-the-art Inception score on CIFAR-10 in the unsupervised settings. Both works suggested a non-trivial idea that the multi-targets information from the reconstruction loss helps to improve the model performance. A closely related work, Boundary Equilibrium GAN (BEGAN) [16] was proposed with a new equilibrium enforcing method. Surprisingly, it demonstrated realistic face generation but is significantly outperformed by DFM on CIFAR-10. This suggests that the traditional adversarial loss remains an important factor to generate realistic complex images.

StackGAN [11] was proposed to overcome the instability issue when training GAN to generate images at higher resolutions (e.g. 256×256). It employed a hierarchical structure by stacking multiple generators that learn to generate images with different resolutions. Their results demonstrated that StackGAN is able to generate appealing images at 256×256 resolution. A different type of hierarchical structure was employed in Karras *et al.* [40] by progressively training different layers in a generator at different stages. As a result of this, they are able to generate high quality images with resolution as high as 1024×1024.

Among latest works, few GAN variants such as CVAE-GAN [47], LSGAN [38], Stacked GAN (SGAN) [57], and Progressive GAN [40] demonstrated their ability in generating high quality images. Qualitatively, their generated images seem to outperform the proposed ArtGAN in terms of subjective image quality. Interestingly, the proposed ArtGAN is able to achieve better Inception score when compared to SGAN [57]. This shows that Inception score [24] is unable to measure the perceptual quality of an image.

A. Conditional Image Synthesis

While unconditional image synthesis is an important research area, many practical applications require the model to be conditioned on some prior information. This prior information has many forms, for instance a distorted image for inpainting [32], [41]; natural image for super-resolution [8] or style transfer [42]–[44]; text codes for text to image translation [10], [11]. Due to the nature of this work, we will only focus on the works related to class-conditioned image generation.

An earlier work that employed conditional setting in GAN was Conditional GAN (CondGAN) [5] where it feeds the labels or modes to the generator and discriminator. However, such setting was demonstrated on less complex images i.e. MNIST and faces [45]. While this website[3] unofficially generated images on CIFAR-10 using CondGAN, the objects in their generated images are hardly recognizable. This is expected because the labels were not fully utilized, as there is no error information backpropagated from the labels. A closed work to ArtGAN is InfoGAN [46] where the discriminator is replaced by a multi-class classifier. Also, InfoGAN has two heads in the discriminator that output c and y separately. Hence, InfoGAN has different architecture compared to ArtGAN. Empirical results showed that InfoGAN is able to learn disentangled representations in an unsupervised manner but the meaning of the representations are uncontrollable during the training stage. As to CondGAN [5], InfoGAN only demonstrated on less complex images, i.e. digits and faces. Bao *et al.* [47] proposed CVAE-GAN that combined Conditional Variational Autoencoder and GAN. CVAE-GAN

[3] http://soumith.ch/eyescream/

is asymmetrically trained by introducing a new objective function for the generator. At the same time, they also trained an encoder network to map the real image to the latent vector. This allows their model to learn a better correlation between the latent vector and the image. They demonstrated that CVAE-GAN is able to generate realistic and diverse images on face, flowers, and birds datasets [20], [21], [48]. However, CVAE-GAN was trained on pre-processed images centered around the objects. Hence, their results are not comparable to ArtGAN as the images used in our experiments are randomly cropped.

In addition to the GAN variants, PixelCNN [33], [34] also demonstrated decent results on conditional image generation but it is computationally expensive for sampling. Built on Deep Generator Network (DGN) [49], Plug and Play Generative Networks (PPGN) [50] is able to produce high quality images at high resolution. It allows different generators and condition networks to be hacked together without having to re-train the generators. However, PPGN differs to the other generative models discussed, herein images are generated in **one-shot** from the latent codes in the traditional generative models. That is to say, in PPGN, images are generated by optimizing the latent codes to produce images that highly activate target neuron in the condition network. The sampling procedure is formalized as an approximate Langevin Markov chain Monte Carlo sampler to ensure diversity. Like other sequential approaches, such gradient-based recursive approach may cause unwanted overhead when deployed in some of the real-world applications, e.g. mobile devices. Nonetheless, they showed that adversarial training is crucial to obtain high quality images.

III. Proposed Method

This section describes the proposed method in detailed. First, we revisit the traditional GAN [1] model. Then, we depict the formulations of the proposed ArtGAN variants. The architecture of the best ArtGAN variant (i.e. ArtGAN-AEM) is depicted in Figure 1.

A. Preliminaries: Generative Adversarial Networks

Generative Adversarial Networks (GAN) [1] contains two networks that are trained by competing with each other. The Generator G aims to generate images $G(\mathbf{z})$ that have a distribution p_G similar to the true data distribution p_{data}, such that $G(\mathbf{z})$ are difficult to differentiate from real images $\hat{\mathbf{x}} \sim p_{data}$. Traditionally, G generates images from some noise vectors $\mathbf{z} \sim p_{noise}$ that are sampled from a distribution p_{noise} (e.g. uniform distribution). On the other hand, the Discriminator D is trained to distinguish the images generated by G from the real images. Overall, the training procedure is a two-player min-max game with the following objective function,

$$\min_G \max_D \mathbb{E}_{\hat{\mathbf{x}}\sim p_{data}}[\log D(\hat{\mathbf{x}})] + \mathbb{E}_{\mathbf{z}\sim p_{noise}}[\log(1 - D(G(\mathbf{z}))] \tag{1}$$

B. ArtGAN

The basic structure of ArtGAN is similar to GAN, such that it consists of a discriminator and a generator that are simultaneously trained using the minmax formulation of GAN, as described in Eq. 1. The key innovation of ArtGAN is to allow feedback from the labels given to each generated image through the loss function. That is, additional label information is fed to the generator to draw a specific subject based on the information, imitating how human learns to draw. This is in contrast to CondGAN [5] that does not fully utilize the labels during training. In order to leverage the labels information, the discriminator is extended to *categorical autoencoder-based discriminator* to output $K + 1$ logistic predictions with K actual categories following the dataset used, and $K+1^{th}$ output as the adversarial class (denoted as Fake category).

Formally, the formulation of a categorical discriminator is written as $D : \mathbb{R}^{H\times W\times C} \rightarrow \mathbb{R}^{K+1}$, where H, W, and C are the height, width, and number of channels of an image, respectively. This is somehow similar to Salimans *et al.* [24], except that the conditional setting is not implemented in their work. While the notations of the *conditional generator* is written as $G : (\mathbf{z}, \bar{c}) \rightarrow \mathbb{R}^{H\times W\times C}$, where $\bar{c}$ is the randomly chosen label for the generated sample in the form of one-hot vector. This allows the generator to learn better from the feedback labels information. Following Salimans *et al.* [24], we modify the categorical discriminator such that D becomes the standard supervised classifier with K outputs, $D : \mathbb{R}^{H\times W\times C} \rightarrow \mathbb{R}^{K}$. Let $l_k(\mathbf{x}) \in D(\mathbf{x})$ be the output of $D(\mathbf{x})$ at class k without activation function and $\mathbf{x}$ is an input image (either from real data or generator). The probability distribution over K classes is given as $p(\mathbf{c}|\mathbf{x})$, such that the predicted probability for each class k is defined as a softmax function,

$$p(c_k|\mathbf{x}) = \frac{e^{l_k}}{\sum_{i=1}^{K} e^{l_i}} \tag{2}$$

The probability distribution function for the binary adversarial prediction $p(y|\mathbf{x})$ of the discriminator is then reformulated as

$$p(y|\mathbf{x}) = \frac{Z(\mathbf{x})}{Z(\mathbf{x}) + 1} \tag{3}$$

where $Z(\mathbf{x}) = \sum_{i=1}^{K} e^{l_i}$. While, $p(y|\mathbf{x}) = 1$ infers that the image $\mathbf{x}$ is real. The benefit of such setting is that the number of parameters can be reduced to relax the over-parametrization problem without changing the output of the softmax, conceptually. The D is then trained by minimizing the following discriminator loss function $\mathcal{L}_D$,

$$\begin{aligned}\mathcal{L}_D = &-\mathbb{E}_{(\hat{\mathbf{x}},\hat{c})\sim p_{data}}\left[\sum_{i=1}^{K} \hat{c}_i \log p(c_i|\hat{\mathbf{x}}) + \log p(y|\hat{\mathbf{x}})\right] \\ &-\mathbb{E}_{\mathbf{z}\sim p_{noise},\bar{c}}\left[\log(1 - p(y|G(\mathbf{z}, \bar{c})))\right]\end{aligned} \tag{4}$$

where $\hat{c}$ is the ground truth one-hot label of the given real image $\hat{\mathbf{x}}$. The generator loss function $\mathcal{L}_G$ to be minimized for training G is defined as,

$$\begin{aligned}\mathcal{L}_G = &-\mathbb{E}_{\mathbf{z}\sim p_{noise},\bar{c}} \\ &\left[\sum_{i=1}^{K} \bar{c}_i \log p(c_i|G(\mathbf{z}, \bar{c})) + \log(p(y|G(\mathbf{z}, \bar{c})))\right]\end{aligned} \tag{5}$$

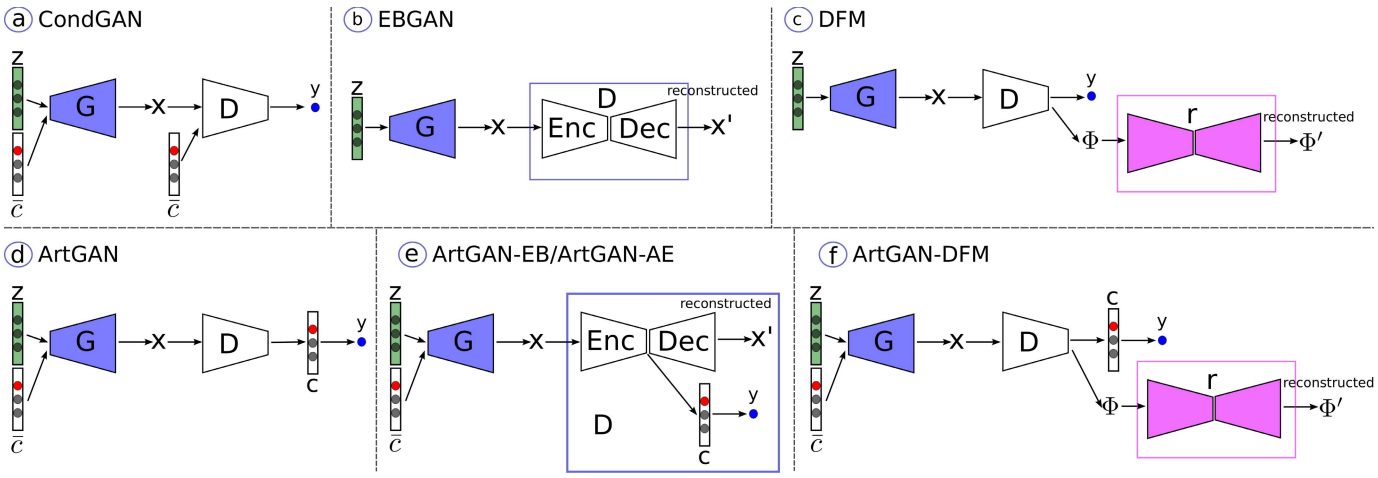


Fig. 3. Different ArtGAN variants (bottom row) compared to the state-of-the-art models (top row). The discriminator in ArtGAN outputs the class predictions and the loss function is computed from the true labels, instead of taking the true labels as input as depicted in CondGAN. Hence, the true labels can be leveraged to train the discriminator and generator. Meanwhile, ArtGAN-EB and ArtGAN-AE share the same model, that is a combination of ArtGAN and EBGAN with shared encoder. However, the decoder in ArtGAN-AE is not trained using the generated samples, as opposed to ArtGAN-EB. ArtGAN-DFM depicts the extension from DFM with conditional settings. Note that InfoGAN has a different architecture compared to ArtGAN since InfoGAN has two heads in the discriminator that output class c and adversarial y predictions separately.

Inspired by recent works [15], [16], [39], we incorporate an autoencoder into the categorical discriminator in ArtGAN for additional complementary information. The core idea of using an autoencoder in the discriminator is that reconstruction-based output offers diverse targets, which produce a very different gradient directions within the minibatch. Conceptually, this improves the efficiency and effectiveness when training a GAN model. Rather than deploying two separate computationally expensive networks (a categorical discriminator and an autoencoder separately), the same architecture and weights are partly shared. In specific, the *encoder* in the autoencoder is shared by the categorical discriminator, as shown in Figure 1. In this paper, the formulations of the categorical autoencoder-based discriminators are described in three different ways. The first two variants, ArtGAN-EB and ArtGAN-AE are implemented using the pixel-level autoencoder, similar to EBGAN [15]. However, these two variants are differed in terms of the discriminator loss functions formulation. The third type, ArtGAN-DFM is an extension of Denoising Feature Matching (DFM) [39] to a conditional setup, forming a *Conditional DFM*. All the ArtGAN variants are summarized in Figure 3 and the details of the loss functions formulations for each of them are described next. Meanwhile, analysis and comparisons between these ArtGAN variants will be discussed in the experimental section.

1) ArtGAN-EB: EBGAN [15] is formulated according to the energy-based models by replacing the discriminator with an autoencoder, such that $D_{AE}(\cdot) = Dec(Enc(\cdot))$, where Dec and Enc are the decoder and encoder, respectively. The discriminator loss $\mathcal{L}_{Deb}$ in EBGAN is given as,

$$\mathcal{L}_{Deb} = \mathbb{E}_{\hat{\mathbf{x}}\sim p_{data}}\Big[||D_{AE}(\hat{\mathbf{x}}) - \hat{\mathbf{x}}||\Big] + \mathbb{E}_{\mathbf{z}\sim p_{noise}}\Big[\max(0, m - ||D_{AE}(G(\mathbf{z})) - G(\mathbf{z})||)\Big] \tag{6}$$

where $||\cdot||$ is a Euclidean norm, and m as a positive margin. The generator loss $\mathcal{L}_{Geb}$ is formulated as,

$$\mathcal{L}_{Geb} = \mathbb{E}_{\mathbf{z}\sim p_{noise}}\Big[||D_{AE}(G(\mathbf{z})) - G(\mathbf{z})||\Big] \tag{7}$$

In order to formulate a conditional energy-based loss function, ArtGAN-EB propose a novel discriminator loss function $\mathcal{L}_{Debc}$ as,

$$\mathcal{L}_{Debc} = \mathcal{L}_D + \mathcal{L}_{Deb} \tag{8}$$

and the new generator loss $\mathcal{L}_{Gae}$ is defined as,

$$\mathcal{L}_{Gae} = \mathcal{L}_G + \mathcal{L}_{Geb} \tag{9}$$

2) ArtGAN-AE: The discriminator loss is similar to ArtGAN-EB (Eq. 8), except that we do not use the generated images as adversarial samples to update the decoder. This was inspired by DFM [39] to use the autoencoder as a source of *complementary information* when updating the generator, instead of using the autoencoder as an adversarial function (as in [15]). Hence, the discriminator loss $\mathcal{L}_{Dae}$ of ArtGAN-AE is formulated as,

$$\mathcal{L}_{Dae} = \mathcal{L}_D + \mathbb{E}_{\hat{\mathbf{x}}\sim p_{data}}\Big[||D_{AE}(\hat{\mathbf{x}}) - \hat{\mathbf{x}}||\Big] \tag{10}$$

Meanwhile, ArtGAN-AE has the same generator loss as ArtGAN-EB (Eq. 9).

3) ArtGAN-DFM: In DFM [39], an additional denoising autoencoder (or denoiser) $r(\cdot)$ is employed to update the generator. The denoiser is trained separately from the discriminator. In specific, the denoiser is trained on the discriminator's hidden state when it is evaluated on the training data. Formally, D is updated according to Eq. 1. Given that $\Phi(\cdot)$ is a hidden state from $D(\cdot)$, the denoiser is trained by minimizing the following loss function $\mathcal{L}_r$,

$$\mathcal{L}_r = \mathbb{E}_{\hat{\mathbf{x}}\sim p_{data}}\Big[||\Phi(\hat{\mathbf{x}}) - r(\Phi(\hat{\mathbf{x}}))||\Big] \tag{11}$$

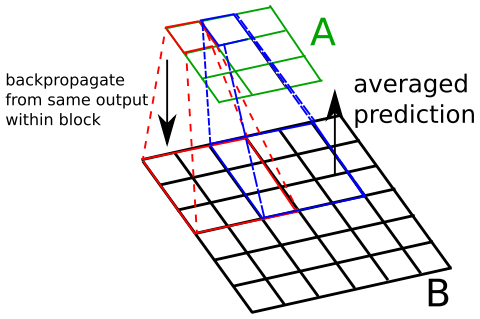


Fig. 4. The proposed strategy using *overlapped average pooling*. Pixels in the same block from B (e.g. $[B_1, \cdots, B_9]$) will vote for an improved pixel in A (e.g. A_1) through averaging.

Then, the generator is trained with the loss function $\mathcal{L}_{Gdfm}$,

$$\mathcal{L}_{Gdfm} = \mathbb{E}_{\mathbf{z}\sim p_{noise}}\Big[\lambda_{denoise}||\Phi(G(\mathbf{z})) - r(\Phi(G(\mathbf{z})))|| - \lambda_{adv}\log D(G(\mathbf{z}))\Big] \tag{12}$$

Warde-Farley and Bengio [39] suggested to fix $\lambda_{adv} = 1$ and set $\lambda_{denoise} = 0.03/n_h$, where n_h is the number of discriminator hidden units fed to the denoiser as input. The modification is straightforward using the categorical discriminator as the discriminator network. Hence, the discriminator loss is same as Eq. 4, and the denoiser loss remains unchanged (Eq. 11). The generator loss $\mathcal{L}_{Gdfmc}$ for the conditional DFM is defined as,

$$\mathcal{L}_{Gdfmc} = \mathbb{E}_{\mathbf{z}\sim p_{noise}}\Big[\lambda_{denoise}||\Phi(G(\mathbf{z})) - r(\Phi(G(\mathbf{z})))||\Big] + \mathcal{L}_G \tag{13}$$

IV. Image Quality (IQ) Strategy

In order to improve the quality of the image generated by ArtGAN, we introduce a novel strategy. The motivation behind this strategy is to generate a set of pixel to vote for an improved (better quality) pixel via average ranking. That is to say, we will train a generator to synthesize images at a resolution $2\times$ higher than the original image size. Then, these generated images will be downsampled by a factor of 2 via average pooling operation as a voting scheme.

In specific, suppose a generator in the traditional GAN trained on a dataset generates 32×32 pixels images, $G : \mathbf{z} \to \mathbb{R}^{32\times32\times C}$, where C is the number of channels. Using IQ strategy, the generator will instead generate 64×64 pixels images (i.e. $2\times$ higher resolution), $G : \mathbf{z} \to \mathbb{R}^{64\times64\times C}$. This is done by adding an upsampling block (typically an upsampling layer followed by a convolutinal layer) between the existing layers in the generator. Then, the generated samples are downsampled, such that $\pi : \mathbb{R}^{64\times64\times C} \to \mathbb{R}^{32\times32\times C}$ where $\pi(\cdot)$ is a downsampling operation. Meanwhile, the input size of the discriminator remains the same as to the original size, such that $D : \mathbb{R}^{32\times32\times C} \to \mathbb{R}^K$, where K is the number of categories.

In this paper, overlapped average pooling is chosen as the downsampling operation. The average pooling operation can be viewed as a form of voting system, as shown in Figure 4. Overlapping the pooling operations discourages the generator from blindly computes the same pixel value within the same pooling block. Overall, when the overlapped average pooling is used, the generator is regularized with two seemingly contradictory constraints: i) the generated pixels within the same pooling block should have similar intensity so that the generated image looks smooth across the same color (e.g. smooth blue sky); ii) the generated pixels must not be naively computed to produce the same intensity that may cause excessive artifacts in the image. During inference, this pooling layer can be removed in order to output higher resolution synthetic images. Readers should be noted that this is different from super-resolution as the nature of this paper focuses on generating random images based on the given labels.

V. Experimental Results

A. Experimental Settings

This section describes the settings that are used in all experiments, unless stated otherwise. All networks are trained with Adam optimizer [51] with an initial learning rate $= 0.0002$, $\beta_1 = 0.5$, and minibatch size $= 100$. The learning rate is decreased by a factor of 10 after iteration $30,000$. Input noise vector $\mathbf{z}$ is a 100-dimensional multivariate random variable sampled using an i.i.d. uniform distributed random generator $U(-1, 1)$. Instance noise [35] is implemented in all discriminators for better training stability. For a fair comparison, we run one gradient descent step for each player in each iteration. Generally, this is better than running more steps of one player than the other [52]. Also in practice, it is very difficult to determine how many steps to use, as the performance is usually inconsistent using the same setting on different datasets. The rest of the settings will be described in the related sections. The experiments were conducted using Tensorflow [53] with one Titan X (Maxwell) GPU.

For evaluation, Inception score is adopted [24] as the quantitative metric. Intuitively, Inception score measures the *objectness* by minimizing entropy per-sample posterior (i.e. each sample is classified with high certainty), as well as the *class diversity* by maximizing the entropy aggregate posterior (i.e. the classifier used in Inception score identifies a wide variety of classes among the samples). However, one should aware that *class diversity* metric becomes meaningless in the conditional setting as the conditional generative models will always generate visually different images in different modes. In addition, the *class diversity* metric can be misleading, i.e. it can be maximized (higher is better) and fooled when the predicted class distributions of all generated samples are uniform. Hence, we split the measurements (*objectness* and *class diversity* metrics) when we report the scores in this paper for performance evaluations.

Since Inception score is calculated by measuring the object class confidence scores, therefore it is not suitable to assess the model performance on artworks. Meanwhile, evaluation of generative model based on the state-of-the-art log-likelihood estimates can be misleading [54]. Hence, the comparative studies are first conducted using the *objectness* metric from Inception score on CIFAR-10 [13] and STL-10 [19] datasets.

Then, Wikiart dataset [23], [55] is used for artworks synthesis based on genres, artists, and styles. Finally, we trained on Oxford-102 [20] and CUB-200 [21] for additional performance assessments.

We used similar design to BEGAN [16] (i.e. employing nearest neighbour upsampling instead of strided deconvolution layer in the generator as suggested by Odena *et al.* [56]) in order to avoid checkerboard artifacts. Between the upsampling layers, there is at least one layer of convolutional layer. The discriminator has the same design as to the traditional GAN with multiple layers of strided convolutional layers. Batch normalization and leaky ReLU are used in both the discriminator and generator. Due to page limit, detailed network descriptions and additional generated samples are available in the appendix. The list of proposed models are as follows:

1) ArtGAN - Baseline model [22].
2) ArtGAN-EB - The first variant of categorical autoencoder-based discriminator.
3) ArtGAN-AE - The second variant of categorical autoencoder-based discriminator.
4) ArtGAN-DFM - The third variant of categorical autoencoder-based discriminator.
5) ArtGAN-M - ArtGAN with **IQ strategy**.
6) ArtGAN-D - It has similar architecture as to ArtGAN-M but without IQ strategy. This model is used to verify if network size is the main factor that contributes to the performance improvements observed on ArtGAN-M.
7) ArtGAN-AEM - ArtGAN-AE with **IQ strategy**.
8) ArtGAN-AEMT - Huang *et al.* [57] employed a trick by updating more steps for the generator per each discriminator update step. Although it is hard to determine number of steps, their setup seems to work well for CIFAR-10. Hence, the same setting is employed in our CIFAR-10 experiment as a comparison.

B. Evaluation and Metric

Evaluation of a generative model is extremely difficult as it is still not clear how to quantitatively evaluate a generative model. This is due to the difficulty in estimating the intractable log-likelihood in many models [54]. The most widely used log-likelihood estimator is the Parzen window estimates [58]. However, Theis *et al.* [54] convincingly argued that this estimator can be quite misleading for high-dimensional data. Recently, Salimans *et al.* [24] proposed Inception score (higher is better) as a different way to assess image quality by using the:

$$\begin{aligned} I(\{\mathbf{x}\}_1^N) &= \exp(\mathbb{E}[D_{KL}(p(y|\mathbf{x})||p(y))]) \\ &\approx \exp(-\mathbb{E}[H(p(y|\mathbf{x}))] + \mathbb{E}[H(\mathbb{E}_{\mathbf{x}}(p(y|\mathbf{x})))]) \end{aligned} \tag{14}$$

where $H(\cdot)$ is the Shannon entropy and $D_{KL}(\cdot)$ is the KullbackLeibler divergence. As aforementioned, this metric measures the *objectness* in the first term (lower is better) and *class diversity* in the second term (higher is better) of the samples. It can be misleading when the *class diversity* metric is fooled. An example can be seen in our experiments when we compare ArtGAN (baseline) and ArtGAN-EB in Table I.

TABLE I

INCEPTION SCORES ON CIFAR-10 EVALUATED AT 32×32 PIXELS. SCORES ARE REPORTED IN THE FORM OF *Mean Score*±*std.* IN THE PROPOSED METHODS COLUMN, THE ITALIC SCORE IS OBJECTNESS METRIC REPORTED IN THE FORM OF *Objectness* (*Class Diversity*)

Model	Scores
Unlabelled	
Infusion training [59]	4.62 ± 0.06
ALI [60] (as reported in [39])	5.34 ± 0.05
BEGAN [16]	5.62
GMAN [61]	6.00 ± 0.19
EGAN-Ent-VI [62]	7.07 ± 0.10
LR-GAN [63]	7.17 ± 0.07
Denoising feature matching [39]	7.72 ± 0.13
Labelled	
SteinGAN [64]	6.35
DCGAN (as reported [64])	6.58
Improved GAN [24]	8.09 ± 0.07
AC-GAN [9]	8.25 ± 0.07
SGAN [57]	8.59 ± 0.12
Proposed methods	
ArtGAN (baseline)	8.21 ± 0.08 *33.24* (*272.90*)
ArtGAN-EB	8.26 ± 0.10 *33.51* (*276.60*)
ArtGAN-AE	8.43 ± 0.09 *31.09* (*262.04*)
ArtGAN-DFM	8.25 ± 0.09 *33.34* (*274.99*)
ArtGAN-M	8.50 ± 0.06 *30.19* (*256.62*)
ArtGAN-D	8.29 ± 0.10 *33.30* (*276.15*)
ArtGAN-AEM	8.53 ± 0.09 ***30.07*** (*256.42*)
ArtGAN-AEMT	$\mathbf{8.81} \pm 0.14$ *30.65* (*269.83*)
Real data	11.24 ± 0.12 *24.32* (*271.76*)

Although ArtGAN-EB performed better than ArtGAN with higher Inception score (ArtGAN-EB = 8.26 compared to ArtGAN = 8.21), it has poor *objectness* score (ArtGAN-EB = 33.51 compared to ArtGAN = 33.24). It shows that the *class diversity* score in ArtGAN-EB has affected the Inception score. This is misleading because the combination of high *class diversity* score and poor *objectness* score implies that the objects in the generated images are hard to recognize. Nonetheless, Inception score is still a preferred metric due to the lack of a better alternative for quantitative measurement. Hence, this paper adopts Inception score but the performance assessment is done mainly based on the *objectness* score since it is a more reliable metric.

In addition, the generated images will be illustrated for visual inspection as human evaluation is always more accurate when accessing the image quality, though can be subjective at times. Furthermore, latent space interpolation is done to "probe" the structure of the latent space $\mathbf{z}$. Qualitatively, the smooth transitions between samples when the latent space is interpolated usually indicates how well the generative models understand the structure of the images.

C. CIFAR-10

CIFAR-10 [13] is a small, well-studied dataset consisting 32×32 pixels RGB images. It is split into 50,000 training

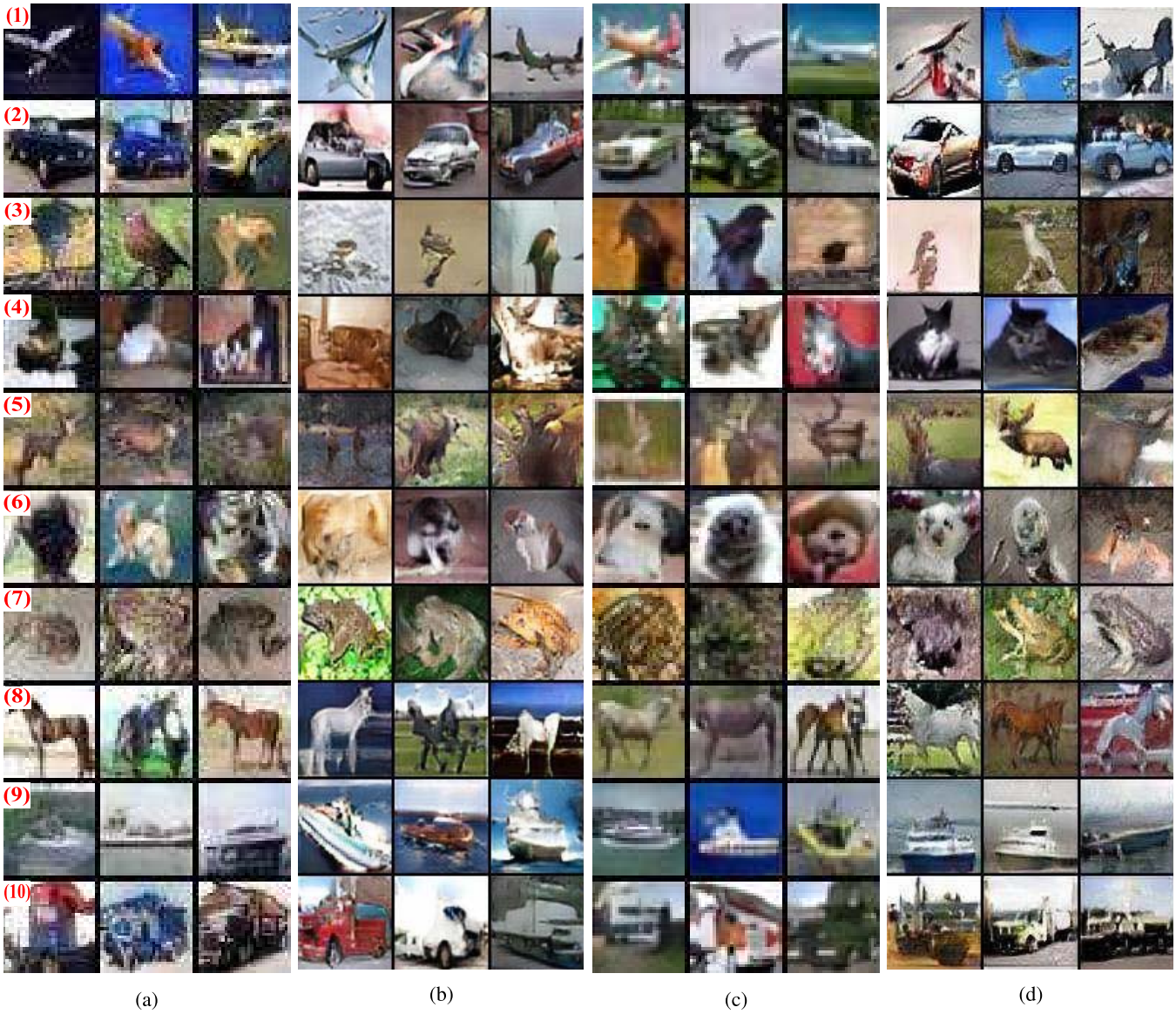


Fig. 5. Comparison of generated CIFAR-10 images with (i.e. ArtGAN-M & ArtGAN-AEM) / without (i.e. ArtGAN & ArtGAN-AE) IQ strategy. Image class from top to bottom: (1) Airplane, (2) Automobile, (3) Bird, (4) Cat, (5) Deer, (6) Dog, (7) Frog, (8) Horse, (9) Ship, (10) Truck. (a) ArtGAN (**32** $\times$ **32**). (b) ArtGAN-M (**64** $\times$ **64**). (c) ArtGAN-AE (**32** $\times$ **32**). (d) ArtGAN-AEM (**64** $\times$ **64**). (Best viewed in color).

images and 10,000 test images from 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck.

All ArtGAN variants are trained on full image size, i.e. 32×32 pixels. When IQ strategy is employed, the generator is able to generate images at a higher resolution (i.e. 64×64). All models are trained for 70,000 iterations and saved every 1,000 iterations. As stated by Gulrajani *et al.* [36], Inception scores of the generative models will continue to oscillate with non-negligible amplitude at convergence. Hence, only the best models found based on the *objectness* score are reported in Table I along with the state-of-the-art results. ArtGAN-AEMT obtains state-of-the-art result with a score of 8.81 ± 0.14, outperformed two latest methods - SGAN [57] (8.59 ± 0.12) and AC-GAN [9] (8.25 ± 0.07). Qualitatively, the proposed models also able to produce many samples with high visual fidelity, especially when IQ strategy is employed as shown in Figure 5. Particularly, the samples drawn by ArtGAN-AEM have finer details, e.g. cats are more recognizable with better ears shape (row 1 and 2), most of the frogs are drawn with clear contour (row 2 and 3), etc.

Interestingly, SGAN [57] demonstrated subjectively better image quality despite lower Inception score when compared to the proposed ArtGAN-AEMT. In SGAN [57], similar loss function is used for their conditional loss, i.e. cross-entropy for labels. Hence, we deduce that network design and training procedure (e.g. training the networks in a hierarchical manner as in SGAN [57] and Progressive GAN [40]) are important factors for achieving better perceptual image quality. Meanwhile, the proposed ArtGAN baseline has lower subjective image quality and Inception score (8.21 ± 0.08) when compared with SGAN. Hence, it is clear that the proposed IQ strategy helps improve the Inception score but not the image quality.

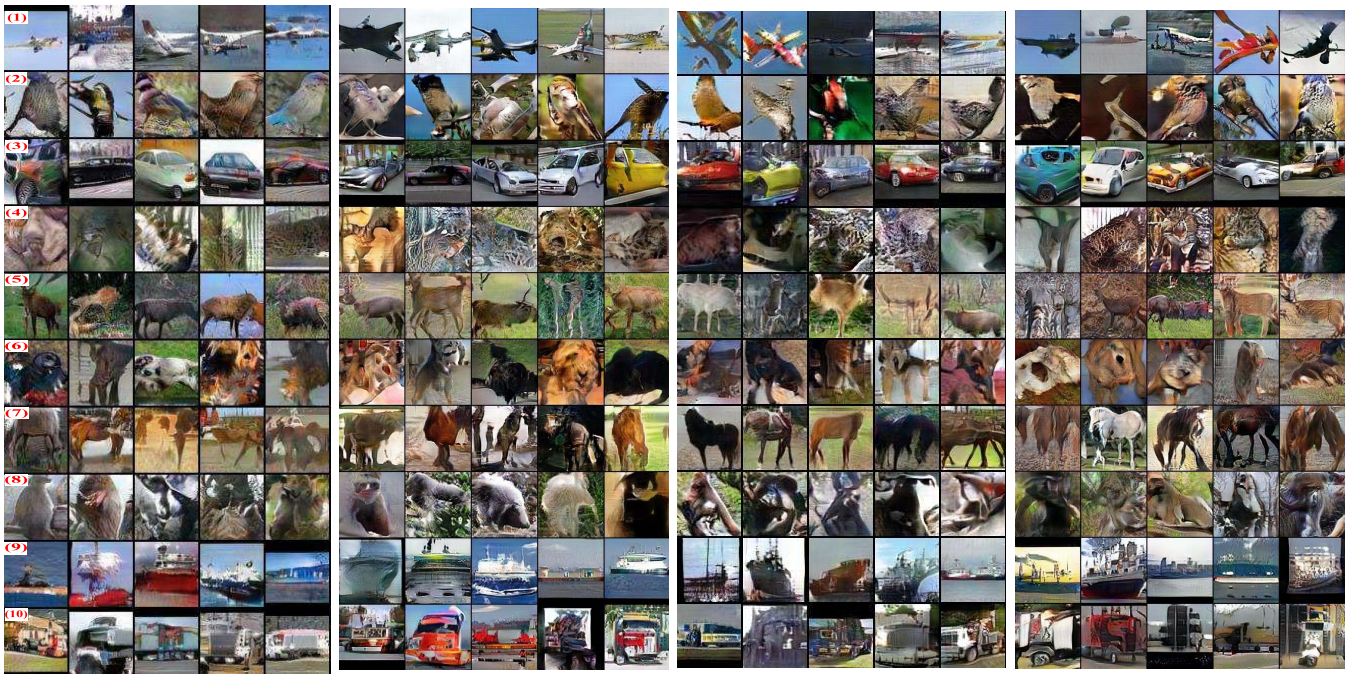


Fig. 6. Comparison of generated STL-10 images with (i.e. ArtGAN-M & ArtGAN-AEM) / without (i.e. ArtGAN & ArtGAN-AE) IQ strategy. Image class from top to bottom: (1) Airplane, (2) Bird, (3) Car, (4) Cat, (5) Deer, (6) Dog, (7) Horse, (8) Monkey, (9) Ship, (10) Truck. (a) ArtGAN (**64** × **64**). (b) ArtGAN-M (**128** × **128**). (c) ArtGAN-AE (**64** × **64**). (d) ArtGAN-AEM (**128** × **128**). (Best viewed in color).

We deduce that the higher feature dimension introduced by IQ strategy complements the loss function by learning richer representations. This encourages the generation of images that are easy to categorize, resulting in higher Inception score.

D. STL-10

STL-10 [19] is a dataset inspired by CIFAR-10 with higher image resolution (i.e. 96 × 96 pixels). However, it contains fewer labelled training examples and has a very large set of unlabelled examples. Although STL-10 is primarily used for unsupervised learning, we employed the dataset for conditional image synthesis in a supervised fashion. In particular, we only employed the labelled examples during training, which contains 5,000 samples from 10 classes: airplane, bird, car, cat, deer, dog, horse, monkey, ship, and truck. As such, it makes STL-10 a more challenging dataset than CIFAR-10.

During training, we randomly cropped 84 × 84 pixels from the 96 × 96 pixels images. Then, the images are resized and trained at 64 × 64 resolution. Meanwhile, the proposed models trained using IQ strategy are able to generate samples at 128 × 128 resolution. All models are trained for 50,000 iterations. Similar to CIFAR-10, models are saved every 1,000 iterations and the scores of the best models are reported. The Inception scores are reported in Table II, while the generated samples are shown in Figure 6. It can be noticed that the synthetic images generated by ArtGAN-AEM trained with IQ strategy are clearer and sharper without much artifacts. For instance, the face features of the dogs are more recognizable (row 1-3). No mode collapse is observed in this experiment.

TABLE II
INCEPTION SCORES ON STL-10 EVALUATED AT 64 × 64 PIXELS. READERS MAY REFER TO TABLE I FOR SCORES DESCRIPTIONS

Model	Scores
ArtGAN (baseline)	9.72 ± 0.14 *31.03 (301.63)*
ArtGAN-EB	9.73 ± 0.12 *30.22 (293.89)*
ArtGAN-AE	9.65 ± 0.08 *31.04 (299.50)*
ArtGAN-DFM	9.63 ± 0.09 *31.25 (300.89)*
ArtGAN-M	$\mathbf{10.12} \pm 0.09$ *29.05 (293.90)*
ArtGAN-D	9.87 ± 0.09 *31.03 (306.39)*
ArtGAN-AEM	10.07 ± 0.09 **28.18**(*283.81*)
Real data	15.48 ± 0.76 *15.04 (232.17)*

E. More Ablation Studies

In order to further understand the effects of different ArtGAN variants, we conduct extensive ablation studies by comparing the performances of the ArtGAN models on CIFAR-10 (Table I) and STL-10 (Table II) datasets. Note that the performances are evaluated based on the *objectness* metric only, unless specified otherwise. Below we summarize our findings.

First, the effectiveness of the IQ strategy can be assessed by comparing ArtGAN-M with the baseline (ArtGAN) and ArtGAN-D. Although ArtGAN-D has more parameters than

the baseline, it does not exhibit overfitting problem since its performance is similar to the baseline. We can notice that ArtGAN-M outperformed the baseline and ArtGAN-D significantly. This shows that the extra convolutional layers in the generator are not the main factor that contribute to the performance improvement when the IQ strategy is employed. This is because both ArtGAN-M and ArtGAN-D have the same number of layers, so it proves that the additional upsampling layer introduced in the IQ strategy is the main reason for the improvement. We deduce that richer representation can be learned with higher feature dimension.

Second, ArtGAN-DFM performed poorer than the baseline. In ArtGAN-DFM, the features fed to the denoiser are extracted from the discriminator that is still in training mode. Hence, we speculate that measuring the loss using these primitive features might cause instability when training the denoiser and generator. Therefore, we encourage to compute the losses by leveraging the true data directly.

Third, inconsistent performance can be noticed in ArtGAN-EB, where it performed best on one dataset but worst on the other. This suggests that additional adversarial loss does not always complement a model. This is because the primitive adversarial samples may provide noisy information that hamper the training process. ArtGAN-AE exhibited more consistent performances with either better or comparable scores. We also trained another variant using only the Energy-based adversarial loss (i.e. traditional adversarial loss is removed). Unfortunately, we found that this model failed to learn, produced collapsed and meaningless images. This deduces that traditional adversarial loss is still a better choice for adversarial training.

Finally, ArtGAN-AEM (ArtGAN-AE with IQ strategy) achieved the best results with consistent and significant improvements. Meanwhile, ArtGAN-AEMT has the best overall Inception score on CIFAR-10 (8.81).

F. Oxford-102

Oxford-102 [20] consists of 102 flower species. Each category has around 40 to 258 samples. The samples have large variations in terms of scale, pose, and light. Beside this, some categories exhibit very similar appearance to each other. The model was trained for 30,000 iterations with learning rate reduced after iteration 15,000. The images were saved at 256×256 resolution. During training, the images are randomly cropped at 224×224, and then resized to 64×64.

Two experiments were conducted. In the first experiment, batch size $= 102$ is used. In the generator, one sample is drawn for each class during the training stage. We found out that the image quality is high but it suffered from mode collapse, i.e. the generated images look almost exactly the same within a class. In the second experiment, 20 classes are randomly chosen in each iteration and with this, 5 samples are drawn for each class during the training stage. This solved the mode collapse problem, suggests that more adversarial images should be sampled for each class in the same iteration to learn more diverse correlations between the latent codes and the image space. Sample of the generated images are depicted in Figure 7. Although the discriminator performed poorly on the classification of flower species ($\sim$ 50% accuracy), Figure 7 shows that ArtGAN-AEM is able to generate high quality flower images that look natural with distinctive species-typical features, i.e. color and shape.

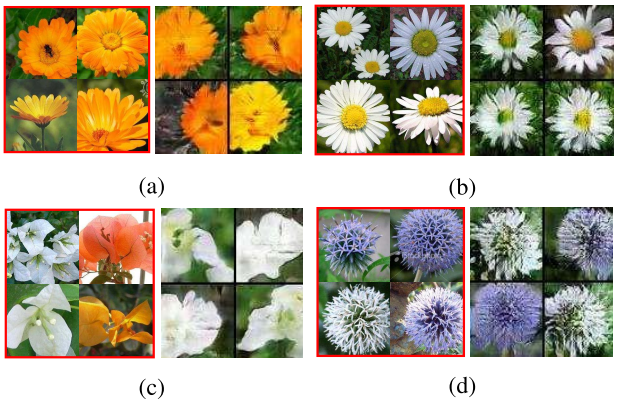


Fig. 7. Sample generated images on Oxford-102 flowers. Left (red box): Groundtruth; Right: Generated samples. (a) Barbeton Daisy. (b) Oxeye Daisy. (c) Globe Thistle. (d) Bougainvillea. (Best viewed in color).

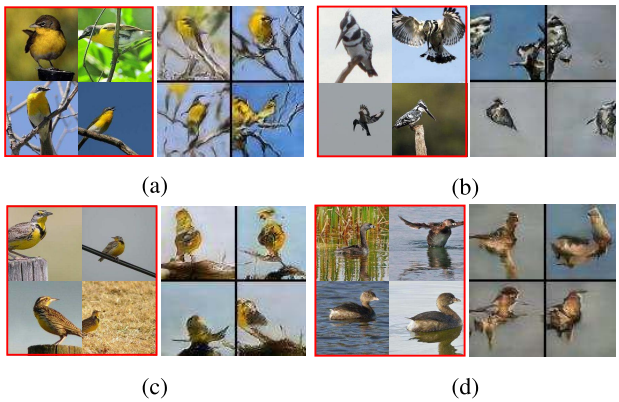


Fig. 8. Sample generated images on CUB-200 birds. Left (red box): Groundtruth; Right: Generated samples. (a) Yellow Breasted Chat. (b) Pled Kingfisher. (c) Pied Billed Grebe. (d) Western Meadowlark. (Best viewed in color).

G. CUB-200

Caltech-UCSD Birds-200-2011 (CUB-200) [21] contains 11,788 samples from 200 bird species. The images are pre-processed in the similar way as to Oxford-102, i.e. model is trained at 64×64 resolution after cropping and resizing.

In order to avoid the mode collapse experienced in Oxford-102, the model herein follows the same settings (i.e. randomly choosing 20 classes in each iteration with 5 samples per class). The generated image samples are shown in Figure 8. Similar to Oxford-102 dataset, the discriminator has a poor performance on the bird species classification ($\sim$20% accuracy). Interestingly, the figures show that ArtGAN-AEM is still able to draw the characteristics of different bird species, e.g. colors, shape, and body size. However, the body structures of the birds are not well-learned.

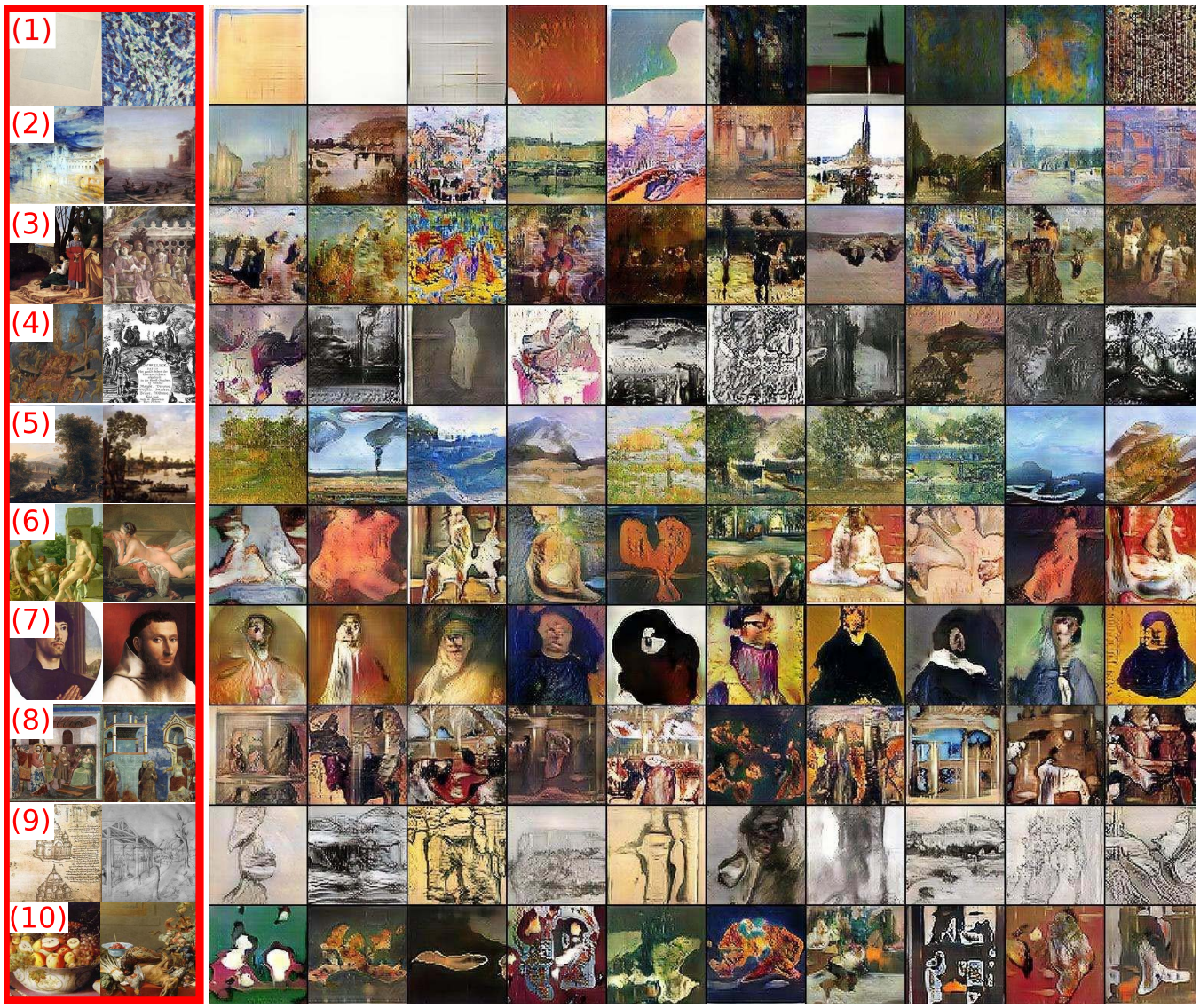


Fig. 9. Wikiart dataset: Generated *genres* images at **128** $\times$ **128** pixels. Images in the red bounding box are groundtruth. Genres class from top to bottom: (1) Abstract, (2) Cityscape, (3) Genre painting, (4) Illustration, (5) Landscape, (6) Nude, (7) Portrait, (8) Religious, (9) Sketch and study, (10) Still life. (Best viewed in color).

H. WikiArt

Wikiart is a fine-art paintings dataset first introduced by Saleh and Elgammal [23]. The paintings were obtained from the wikiart.org website. Currently, Wikiart is the largest public dataset available that contains around 80,000 annotated paintings for genres, artists and styles classification tasks. However, not all paintings are used in all tasks. To be specific, all paintings are used for 27 *styles* classification. But, there are only 60,000 paintings annotated for 10 *genres*, and only around 20,000 paintings are annotated for 23 *artists*. In this paper, we used an extended version of Wikiart dataset.[4] The extended dataset is randomly split to training and test sets for a fair comparison.

The Wikiart images were prepared in 256 $\times$ 256 resolution. In this paper, however, at each iteration of the training stage, we randomly cropped the images into 224 $\times$ 224 resolution. Since the proposed models are built on standard GAN, we experienced similar problem found in [11]. That is, the proposed models are prone to generate nonsensical images when trained using 128 $\times$ 128 or higher resolutions images. As a result of that, we resized the cropped images to 64$\times$64 resolution. Three different ArtGAN-AEM models were trained for different tasks (i.e. *styles*, *genres* and *artists*) for 50,000 iterations. The results are reported using the final model (i.e. model at iteration 50,000). In general, it is observed that ArtGAN-AEM is able to learn artistic representations and generate high quality paintings. Detailed discussions are as follows:

1) Genre: The generated paintings based on genre are showed in Figure 9. Out of the three tasks, genres classification can be considered as the most easiest task [55]. Hence, it is expected that ArtGAN-AEM is able to draw many meaningful paintings based on the genre. For instance, one should be able to differentiate *abstract paintings*, *cityscape*, *landscape*, and *portraits* from other classes easily. The synthesized paintings show that ArtGAN-AEM is able to recognize and draw high

[4]https://github.com/cs-chan/ArtGAN

Fig. 10. Wikiart dataset: Generated *artists* images at **128** $\times$ **128** pixels. Images in the red bounding boxes are groundtruth. Artists class from left to right, **Top**: (1) Albrecht Durer, (2) Boris Kustodiev, (3) Camille Pissarro, (4) Childe Hassam, (5) Claude Monet, (6) Edgar Degas, (7) Eugene Boudin, (8) Gustave Dore, (9) Ilya Repin, (10) Ivan Aivazovsky, (11) Ivan Shishkin, (12) John Singer Sargent; **Bottom**: (13) Marc Chagall, (14) Martiros Saryan, (15) Nicholas Roerich, (16) Pablo Picasso, (17) Paul Cezanne, (18) Pierre Auguste Renoir, (19) Pyotr Konchalovsky, (20) Raphael Kirchner, (21) Rembrandt, (22) Salvador Dali, (23) Vincent van Gogh. (Best viewed in color).

quality paintings on these genres. An interesting observation can also be observed in the *genre painting* (i.e. No 3). Not to be confused with "genre", "genre paintings" is a pictorial representation of scenes or events from everyday life, such as markets, parties, etc. Hence, a group of people is usually visible in this type of paintings. Figure 9 shows that ArtGAN-AEM is able to draw several human-like figures in a few synthetic paintings (i.e. column 5, 6 and 10 of No 3). The model may not be able to understand the true meaning of *genre paintings*, but this observation shows that ArtGAN-AEM is able to find certain semantic cues.

2) Artist: Figure 10 shows the synthetic paintings based on artist. Learning visual representations in this task is possible as artists usually have their own preferences when deciding what to draw, what kind of styles to use, etc. Hence, many visual similarities can be found from those artworks within the same artist. For example, this can be seen in the paintings of *Nicholas Roerich*. He is a Russian who settled in

Fig. 11. Wikiart dataset: Generated *styles* images at **128** $\times$ **128** pixels. Images in the red bounding boxes are groundtruth. Styles class from top to bottom, **Left**: (1) Abstract Expressionism, (2) Action painting, (3) Analytical Cubism, (4) Art Nouveau, (5) Baroque, (6) Color Field Painting, (7) Contemporary Realism, (8) Cubism, (9) Early Renaissance; **Middle**: (10) Expressionism, (11) Fauvism, (12) High Renaissance, (13) Impressionism, (14) Mannerism Late Renaissance. (15) Minimalism, (16) Naive Art Primitivism, (17) New Realism, (18) Northern Renaissance; **Right**: (19) Pointillism, (20) Pop Art, (21) Post Impressionism, (22) Realism, (23) Rococo, (24) Romanticism, (25) Symbolism, (26) Synthetic Cubism, (27) Ukiyo-e. (Best viewed in color).

Himachal Pradesh, India (a mountainous state) for a long time. As a result of that, many of his famous masterpieces depict the beauty of the mountains with expressive colors and fluid brushwork. These characteristics appear in all the synthesized paintings of *Nicholas Roerich* (i.e. No 15). At the same time, all the synthesized paintings of *Gustave Dore* (i.e. No 8) also clearly display his primary approach in engraving, etching, and lithography, which result in grayish artworks. However, the synthesized paintings conditioned on *Vincent van Gogh* appear to be colourless (i.e. No 23). After some investigations, we found an interesting fact that more than half of his artworks were annotated as *sketch and study* genre in the Wikiart dataset. Among all his artworks, most *Van Goghs* palette consisted mainly of sombre earth tones, particularly dark brown, and show no sign of the vivid colours that distinguish from his later work, e.g. the famous The Starry Night masterpiece. This explains the behaviour of the trained model. But, this is still not competent as the striking colour, emphatic brushwork, and the contoured forms of his work that powerfully influenced the Expressionism style in modern art is not well-learned by ArtGAN. *Eugene Boudin* is a marine painter and he has always favoured rendering the sea and along its shores in his artworks. Meanwhile, *Ivan Shishkin* became famous for his forest landscapes. All these preferences can be visualize in all the synthesized paintings of *Eugene Boudin* (i.e. No 7) and *Ivan Shishkin* (i.e. No 11), respectively.

3) Style: Synthetic paintings based on style are shown in Figure 11. Out of the three tasks, style is the most difficult task. For instance, as highlighted in Section II, it is hard to recognize *Renaissance* art. Beside that, it is also a very challenging task to differentiate *Baroque* and *Rococo* as they are historically related. Generally, they are differentiated by the "feelings" they give to their viewers (i.e. curator). *Baroque* art often depicts violence, darkness, and the nudes are more plump compared to the *Rococo* artwork. During mid-1700s, artists gradually moved away from *Baroque* into the modern *Rococo* style. *Rococo* art was often light-hearted, pastoral, and a rosy-tinted view of the world. A subjective observation can be seen in Figure 11 such that *Baroque* synthetic arts (i.e. No 5) are drawn using darker color than the *Rococo* counterparts (i.e. No 23). The color intensity shows that ArtGAN-AEM has managed to learn some of these characteristics. Meanwhile, *Ukiyo-e* is a type of Japanese art flourished from the 17^{th} through 19^{th} centuries. It is produced using the woodblock printing for mass production and a large portion of these paintings appear to be yellowish due to the paper material. It is observed that such characteristics are generated in the synthetic Ukiyo-e style paintings (i.e. No 27).

I. Latent Space Interpolation

In this section, we demonstrate that ArtGAN is not simply memorizing the training data, but can truly generate

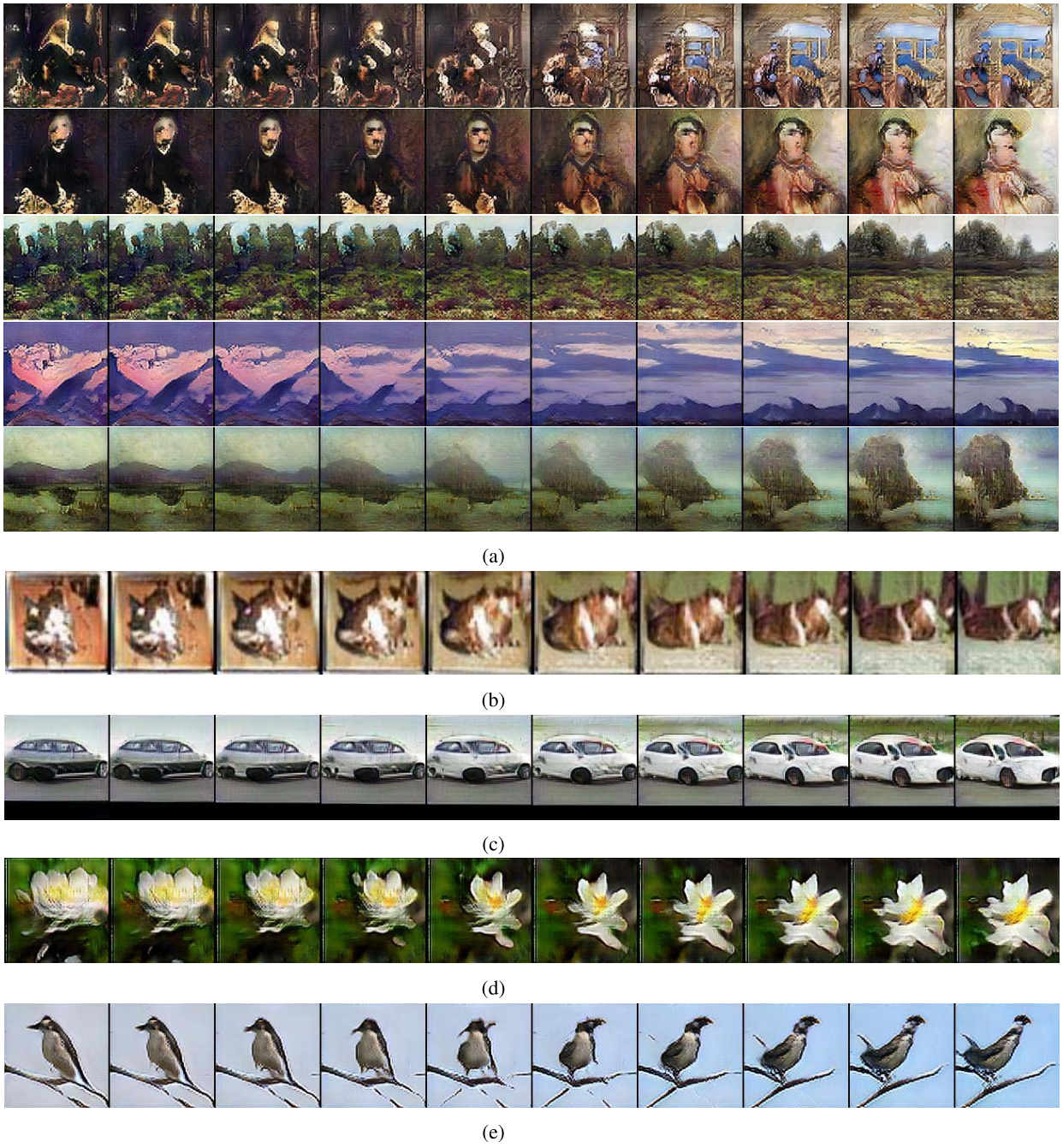


Fig. 12. Interpolations over the latent space **z** in (a) Wikiart, (b) CIFAR-10, (c) STL-10, (d) Oxford-102 and (e) CUB-200 datasets. It demonstrates that ArtGAN does not memorize the training data since samples show smooth transitions and each image looks plausible. (Best viewed in color).

novel images. Walking on the manifold of the latent space **z** can examines the signs of memorization, i.e. sharp image transitions along the latent space indicate high probability that the model memorizes the true data space. This will be an undesired property as it also implies that the relation between the latent codes and image space is not well learned. Figure 12 shows that the generated samples have smooth semantic changes and look plausible. For instance, the bird in the synthetic images of CUB-200 rotated from left to right smoothly. This confirms that ArtGAN is not memorizing and has learned relevant, interesting, and rich visual representations.

VI. Conclusion

This paper proposed a novel GAN variant called ArtGAN which leverages the labels information for better learning representation and image quality. Empirically, it showed that an extension of ArtGAN (i.e. ArtGAN-AEM) achieved state-of-the-art results on CIFAR-10 and STL-10. Furthermore, ArtGAN-AEM showed the superiority in generating high quality and plausibly looking images on Oxford-102 and CUB-200 datasets. Not to mention, the generated paintings showed that ArtGAN-AEM is able to learn artistic representations from the Wikiart paintings that are usually non-figurative and abstract.

For future work, we are looking forward to extend the work for other interesting applications, such as natural to artistic image translation based on a desired semantic-level mode, e.g. *style*.

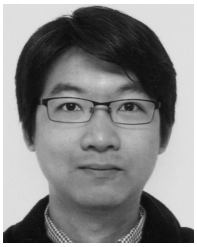
ACKNOWLEDGMENT

The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

REFERENCES

[1] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[2] K. Simonyan and A. Zisserman. (Sep. 2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[3] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[5] M. Mirza and S. Osindero. (2014). "Conditional generative adversarial nets." [Online]. Available: https://arxiv.org/abs/1411.1784
[6] A. Radford, L. Metz, and S. Chintala. (2015). "Unsupervised representation learning with deep convolutional generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06434
[7] J. T. Springenberg. (2015). "Unsupervised and semi-supervised learning with categorical generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06390
[8] E. L. Denton *et al.*, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1486–1494.
[9] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2642–2651.
[10] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1060–1069.
[11] H. Zhang *et al.*, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5908–5916.
[12] Y. LeCun, C. Cortes, and C. J. Burges, "The mnist database of handwritten digits," New York Univ., New York, NY, USA, Tech. Rep., 1998.
[13] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.
[14] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[15] J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–12.
[16] D. Berthelot, T. Schumm, and L. Metz. (Mar. 2017). "BEGAN: Boundary equilibrium generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1703.10717
[17] Y. Wang, L. Zhang, and J. van de Weijer. (Dec. 2016). "Ensembles of generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1612.00991
[18] G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft, and K. Q. Weinberger, "Snapshot ensembles: Train 1, get *M* for free," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14.
[19] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 215–223.
[20] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. Indian Conf. Comput. Vis., Graph. Image Process.*, Dec. 2008, pp. 722–729.
[21] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD birds-200-2011 dataset," Dept. Comput., Neural Syst., California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2011-001, 2011.
[22] W. R. Tan, C. S. Chan, H. E. Aguirre, and K. Tanaka, "ArtGAN: Artwork synthesis with conditional categorical GANs," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2017, pp. 3760–3764.
[23] B. Saleh and A. Elgammal. (2015). "Large-scale classification of fine-art paintings: Learning the right metric on the right feature." [Online]. Available: https://arxiv.org/abs/1505.00855
[24] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2226–2234.
[25] V. Mnih and G. E. Hinton, "Generating more realistic images using gated MRF's," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 2002–2010.
[26] N. Le Roux, N. Heess, J. Shotton, and J. Winn, "Learning a generative model of images by factoring appearance and shape," *Neural Comput.*, vol. 23, no. 3, pp. 593–650, 2011.
[27] H. Shim, "Probabilistic approach to realistic face synthesis with a single uncalibrated image," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3784–3793, Aug. 2012.
[28] D. P. Kingma and M. Welling. (Dec. 2013). "Auto-encoding variational Bayes." [Online]. Available: https://arxiv.org/abs/1312.6114
[29] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3581–3589.
[30] Y. Bengio, L. Yao, G. Alain, and P. Vincent, "Generalized denoising auto-encoders as generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 899–907.
[31] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. Wierstra, "DRAW: A recurrent neural network for image generation," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1462–1471.
[32] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1747–1756.
[33] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, and A. Graves, "Conditional image generation with PixelCNN decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4790–4798.
[34] T. Salimans, A. Karpathy, X. Chen, and D. P. Kingma. (2017). "PixelCNN++: Improving the PixelCNN with discretized logistic mixture likelihood and other modifications." [Online]. Available: https://arxiv.org/abs/1701.05517
[35] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–24.
[36] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5767–5777.
[37] G.-J. Qi. (2017). "Loss-sensitive generative adversarial networks on Lipschitz densities." [Online]. Available: https://arxiv.org/abs/1701.06264
[38] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2813–2821.
[39] D. Warde-Farley and Y. Bengio, "Improving generative adversarial networks with denoising feature matching," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–11.
[40] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–26.
[41] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2536–2544.
[42] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5967–5976.
[43] N. Wang, X. Gao, L. Sun, and J. Li, "Bayesian face sketch synthesis," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1264–1274, Mar. 2017.
[44] M. Elad and P. Milanfar, "Style transfer via texture synthesis," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2338–2351, May 2017.
[45] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class Project Stanford CS231N, Convolutional Neural Netw. Vis. Recognit., Winter Semester*, vol. 2015, no. 5, pp. 1–9, 2015.
[46] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2172–2180.
[47] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "CVAE-GAN: Fine-grained image generation through asymmetric training," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2764–2773.
[48] G. B. Huang and E. Learned-Miller, "Labeled faces in the wild: Updates and new reporting procedures," Univ. Massachusetts, Amherst, MA, USA, Tech. Rep. UM-CS-2014-003, May 2014.
[49] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, and J. Clune, "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3387–3395.

[50] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 4467–4477.
[51] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[52] I. Goodfellow. (Dec. 2016). "Nips 2016 tutorial: Generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1701.00160
[53] M. Abadi *et al.* (2016). "TensorFlow: Large-scale machine learning on heterogeneous distributed systems." [Online]. Available: https://arxiv.org/abs/1603.04467
[54] L. Theis, A. van den Oord, and M. Bethge. (2015). "A note on the evaluation of generative models." [Online]. Available: https://arxiv.org/abs/1511.01844
[55] W. R. Tan, C. S. Chan, H. E. Aguirre, and K. Tanaka, "Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2016, pp. 3703–3707.
[56] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, pp. 1–8, 2016. [Online]. Available: http://distill.pub/2016/deconv-checkerboard
[57] X. Huang, Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie, "Stacked generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1866–1875.
[58] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.
[59] F. Bordes, S. Honari, and P. Vincent, "Learning to generate samples from noise through infusion training," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–19.
[60] V. Dumoulin *et al.*, "Adversarially learned inference," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–18.
[61] I. Durugkar, I. Gemp, and S. Mahadevan, "Generative multi-adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14.
[62] Z. Dai, A. Almahairi, P. Bachman, E. Hovy, and A. Courville, "Calibrating energy-based generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–17.
[63] Y. Jianwei, K. Anitha, B. Dhruv, and P. Devi, "LR-GAN: Layered recursive generative adversarial networks for image generation," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–21.
[64] D. Wang and Q. Liu. (2016). "Learning to draw samples: With application to amortized mle for generative adversarial learning." [Online]. Available: https://arxiv.org/abs/1611.01722
[65] T. S. Eliot, *The Sacred Wood: Essays on Poetry and Criticism*. New York, NY, USA: Alfred A. Knopf, 1921.



Wei Ren Tan received the bachelor's and master's degrees in computer science from the University of Malaya, Kuala Lumpur, Malaysia, in 2010 and 2013, respectively, and the Doctor of Engineering degree from Shinshu University, Japan, in 2017. He is currently a Post-Doctoral Research Fellow with the Computer and Communication Research Center, National Tsing Hua University, Taiwan. His research interests include computer vision, machine learning, and deep learning, focusing on image and video analysis.

Chee Seng Chan (S'05–M'09–SM'14) received the Ph.D. degree from the University of Portsmouth, U.K., in 2008. He is currently an Associate Professor with the Faculty of Computer Science and Information Technology, University of Malaya, Malaysia. His research interests include computer vision and fuzzy set theory, particularly on image/video content analysis. He received several notable awards, such as the Young Scientist Network-Academy of Sciences Malaysia in 2015 and the Hitachi Research Fellowship in 2013. He is/was the Founding Chair of the IEEE Computational Intelligence Society Malaysia Chapter, the Organizing Chair of the Asian Conference on Pattern Recognition (ACPR2015), and the General Chair of the IEEE International Workshop on Multimedia Signal Processing (MMSP2019) and the IEEE Visual Communications and Image Processing (VCIP2013).

Hernán E. Aguirre received the Engineer degree in computer systems from Escuela Politécnica Nacional, Quito, Ecuador, in 1992, and the M.S. and Ph.D. degrees from Shinshu University, Nagano, Japan, in 2000 and 2003, respectively. He is currently an Associate Professor with Shinshu University. He has authored over 130 international journal and conference research papers in related areas. His research interests include evolutionary computation, multidisciplinary design optimization, and sustainability. Dr. Aguirre is a member of the IEICE and IPSJ.

Kiyoshi Tanaka (M'95) received the B.S. and M.S. degrees in electrical engineering and operations research from National Defense Academy, Yokosuka, Japan, in 1984 and 1989, respectively, and the Dr. Eng. degree from Keio University, Tokyo, Japan, in 1992. He is currently a Full Professor at Shinshu University, Nagano, Japan. He is also the Director of the Global Education Center and the Vice-President of Shinshu University. His research interests include image and video processing, 3D point cloud processing, information hiding, human visual perception, evolutionary computation, multi-objective optimization, smart grid, and their applications. He is a fellow of IIEEJ and a member of IEICE, IPSJ, and JSEC.