ARTGAN: ARTWORK SYNTHESIS WITH CONDITIONAL CATEGORICAL GANS

Wei Ren Tan⋆ *Chee Seng Chan*‡ *Hernán E. Aguirre*⋆ *Kiyoshi Tanaka*⋆

⋆Faculty of Engineering, Shinshu University, Nagano, Japan
‡Centre of Image & Signal Processing, Fac. Comp. Sci. & Info. Tech., University of Malaya, Malaysia
{*14st203c@shinshu-u.ac.jp; cs.chan@um.edu.my; ahernan@shinshu-u.ac.jp; ktanaka@shinshu-u.ac.jp*}

Fig. 1: Artwork Generation: Comparison between DCGAN (top), GAN/VAE (middle), and ARTGAN (bottom)

ABSTRACT

This paper proposes an extension to the Generative Adversarial Networks (GANs), namely as ARTGAN to synthetically generate more challenging and complex images such as artwork that have abstract characteristics. This is in contrast to most of the current solutions that focused on generating natural images such as room interiors, birds, flowers and faces. The key innovation of our work is to allow back-propagation of the loss function w.r.t. the labels (randomly assigned to each generated images) to the generator from the discriminator. With the feedback from the label information, the generator is able to learn faster and achieve better generated image quality. Empirically, we show that the proposed ARTGAN is capable to create realistic artwork, as well as generate compelling real world images that globally look natural with clear shape on CIFAR-10.

Index Terms— image synthesis, generative adversarial networks, deep learning

1. INTRODUCTION

"I paint objects as I think them, not as I see them"
– Pablo Picasso

Recently, Generative Adversarial Networks (GANs) [3,5, 10,13] have shown significant promise in synthetically generate natural images using the MNIST [9], CIFAR-10 [7], CUB-200 [19] and LFW datasets [6]. However, we could notice that all these datasets have some common characteristics: i) Most of the background/foreground are clearly distinguishable; ii) Most of the images contain only one object per image and finally iii) Most of the objects have fairly structured shape such as numeric, vehicles, birds, face etc.

In this paper, we would like to investigate if machine can create (more challenging) images that do not exhibit any of the above characteristics, such as the artwork depicted in Fig. 1. Artwork is a mode of creative expression, coming in different kinds of forms, including drawing, naturalistic, abstraction, etc. For instance, artwork can be neither non-figurative nor representable, e.g *Abstract* paintings. Therefore, it is very hard to understand the background/foreground in the artwork. In addition, some artwork do not follow natural shapes, e.g *Cubism* paintings. In the philosophy of art, aesthetic judgement is always applied to artwork based on one's sentiment and taste, which shows one's appreciation of beauty.

An artist teacher wrote an online article [4] and pointed out that an effective learning in art domain requires one to focus on a particular type of skills (e.g practice to draw a particular object or one kind of movement) at a time. Meanwhile, the learning in GANs only involves unlabeled data that doesn't necessarily reflect on a particular subject. In order to imitate such learning pattern, we propose to train GANs focuses on a particular subject by inputting some additional information to it. A similar approach is the Conditional GANs (CondGAN) [10]. The work feed a vector $\vec{y}$ into D and G as an additional input layer. However, there is no feedback from $\vec{y}$ to the intermediate layers. A natural extension is to train D as a classifier with respect to $\vec{y}$ alike to the Categorical GANs (CatGAN) [15] and Salimans et al. [13]. In the former, the work extended D in GANs to K classes, instead of a binary

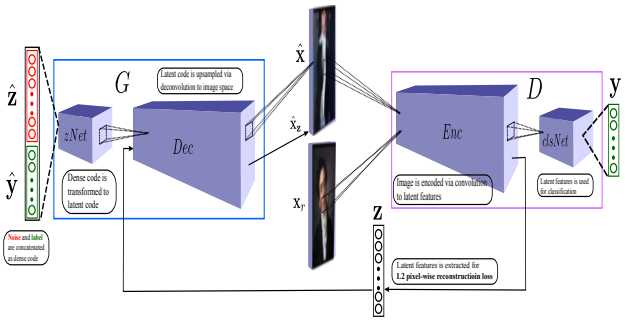

Fig. 2: The overall architecture of ARTGAN. The overall design is similar to the standard GANs except that we have additional input labels $\hat{\mathbf{y}}$ for G and D outputs probability distribution of labels. A connection is also added from Enc to Dec to reconstruct image for L2 pixel-wise reconstruction loss.

output. Then, they trained the CatGAN by either minimize or maximize the Shannon entropy to control the uncertainty of D. In the latter, the work proposed a semi-supervised learning framework and used $K+1$ classes with an additional FAKE class. An advantage of such design is that it can be extended to include more (adversarial) classes, e.gIntrospective Adversarial Networks (IAN) [2] used a ternary adversarial loss that forces D to label a sample as reconstructed in addition to real or fake. However, such work do not use the information from the labels to train G.

To this end, we propose a novel adversarial network namely as ARTGAN that is close to CondGAN [10] but it differs in such a way that we feed $\vec{y}$ to G only and back-propagate errors to G. This allows G to learn better by using the feedback information from the labels. At the same time, ARTGAN outputs $K+1$ classes in D as to the [15] but again we differ in two ways: First, we set a label to each generated images in D based on $\vec{y}$. Secondly, we use sigmoid function instead of softmax function in D. This generalizes the ARTGAN architecture so that it can be extended to other works, e.g multi-labels problem [1], open set recognition problem [14], etc. Inspired by Larsen et al. [8], we also added the L2 pixel-wise reconstruction loss along with the adversarial loss to train G in order to improve the quality of the generated images. Empirically, we show qualitatively that our model is capable to synthesize descent quality artwork that exhibit for instance famous artist styles such as Vincent van Vogh (Fig. 3b). At the same time, our model is also able to create samples on CIFAR-10 that look more natural and contain clear object structures in them, compared to DCGAN [11] (Fig. 5).

2. APPROACH

2.1. Preliminaries

The GANs framework [5] was established with two competitors, the Generator G and Discriminator D. The task of D is to distinguish the samples from G and training data. While, G is to confuse D by generating samples with distribution close to the training data distribution. The GANs objective function is given by:

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

where D is trained by maximizing the probability of the training data (first term), while minimizing the probability of the samples from G (second term).

2.2. ARTGAN

The basic structure of ARTGAN is similar to GANs: it consists of a discriminator and a generator that are simultaneously trained using the minmax formulation of GANs, as described in Eq. 1. The key innovation of our work is to allow feedback from the labels given to each generated image through the loss function in D to G. That is, we feed additional (label) information $\hat{\mathbf{y}}$ to the GANs network to imitate how human learn to draw. This is almost similar to the CondGAN [10] which is an extension of the GANs in which both D and G receive an additional vector of information $\hat{\mathbf{y}}$ as input. That is, $\hat{\mathbf{y}}$ encodes the information of either the attributes or classes of the data to control the modes of the data to be generated. However, it has one limitation as the information of $\hat{\mathbf{y}}$ is not fully utilized through the back-propagation process to improve the quality of the generated images. Therefore, a natural refinement is to train D as a classifier with respect to $\hat{\mathbf{y}}$. To this end, we modify D to output probability distribution of the labels, as to CatGAN [15] except that we set a label to each generated images in D based on $\hat{\mathbf{y}}$ and use cross entropy to back-propagate the error to G. This allows G to learn better by using the feedback information from the labels. Conceptually, this step not only help in speeding up the training process, but also assists the ARTGAN to grasp more abstract concepts, such as artistic styles which are crucial when generating fine art paintings. Also, we use sigmoid function instead of softmax function in D, and employ an additional L2 pixel-wise reconstruction loss as to Larsen et al. [8] along with adversarial loss to improve the training stability. Contrast to Larsen et al. [8], in ARTGAN architecture, the Decoder D shares the same network with Encoder Enc only.

2.3. Details and Formulation of Architecture

Fig. 2 depicts the overall architecture of the proposed ARTGAN. Formally, D maps an input image $\mathbf{x}$ to a probability distribution $p(\mathbf{y}|\mathbf{x})$, $D : \mathbf{x} \rightarrow p(\mathbf{y}|\mathbf{x})$. Generally, D can be separated into two parts: an encoder Enc that produces a latent feature $\mathbf{z}$ followed by a classifier $clsNet$. Similarly, G is fed with a random vector $\hat{\mathbf{z}} \in \mathbb{R}^d \sim \mathcal{N}(0,1)^d$ concatenated with the label information $\hat{\mathbf{y}}$ and outputs a generated image $\hat{\mathbf{x}}$, such that $G : [\hat{\mathbf{z}}, \hat{\mathbf{y}}] \rightarrow \hat{\mathbf{x}}$. G composes of a $zNet$ that

transforms the input to a latent space, followed by a decoder Dec. In this context, $\mathcal{N}(0,1)$ is a normal-distributed random number generator with mean 0 and standard deviation of 1, and d is the number of elements in $\hat{\mathbf{z}}$.

Given K labels and $K+1$ representing the FAKE class, $\mathbf{y}_k$ and $\hat{\mathbf{y}}_{\hat{k}}$ are denoted as one-hot vectors, such that $\mathbf{y} = [y_1, y_2, \ldots, y_{K+1}]$ and $\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_K]$, where $y_k, \hat{y}_{\hat{k}} = 1$ and $y_{i \setminus k}, \hat{y}_{j \setminus \hat{k}} = 0$, $i = \{1, 2, \ldots, K+1\}$, $j = \{1, 2, \ldots, K\}$ when k and $\hat{k}$ are the true classes of the real and generated images, respectively. Then, the data draw from the real distribution is denoted $(\mathbf{x}_r, k) \sim p_{data}$, where $k \in \mathbf{K} = \{1, 2, \ldots, K\}$ is the label of $\mathbf{x}_r$. Meanwhile, p_{noise} is the noise distribution for $\hat{\mathbf{z}}$. For simplicity, we use $G(\hat{\mathbf{z}}, \hat{\mathbf{y}}_{\hat{k}})$ to express the output of G, such that $\hat{k}$ is randomly chosen. Hence, we can minimize the loss function, $\mathcal{L}_D$ w.r.t parameters θ_D in D to update D:

$$\begin{aligned}\mathcal{L}_D = &- \mathbb{E}_{(\mathbf{x}_r,k)\sim p_{data}}\Big[y_k \log p(y_k|\mathbf{x}_r) + \sum_{i\neq k} \log(1 - p(y_i|\mathbf{x}_r))\Big] \\ &- \mathbb{E}_{\hat{\mathbf{z}}\sim p_{noise}, \hat{k}\sim \mathbf{K}}\Big[\log p(y_{K+1}|G(\hat{\mathbf{z}}, \hat{\mathbf{y}}_{\hat{k}})) \\ &+ \sum_{i<K+1} \log(1 - p(y_i|G(\hat{\mathbf{z}}, \hat{\mathbf{y}}_{\hat{k}})))\Big] \end{aligned} \quad (2)$$

Meanwhile, we maximize $\mathcal{L}_D$ to update parameters θ_G in G in order to compete with D. Hence, we can reformulate Eq. 2 as a minimization problem $\mathcal{L}_{adv}$:

$$\begin{aligned}\mathcal{L}_{adv} = &- -\mathbb{E}_{\hat{\mathbf{z}}\sim p_{noise}, \hat{k}\sim \mathbf{K}}\Big[\log p(y_{\hat{k}}|G(\hat{\mathbf{z}}, \hat{\mathbf{y}}_{\hat{k}})) \\ &+ \sum_{i\neq \hat{k}} \log(1 - p(y_i|G(\hat{\mathbf{z}}, \hat{\mathbf{y}}_{\hat{k}})))\Big] \end{aligned} \quad (3)$$

In order to improve the training stability in the ARTGAN, we added the L2 pixel-wise reconstruction loss $\mathcal{L}_{L2}$ along with $\mathcal{L}_{adv}$. Given the latent feature $\mathbf{z}$ output from Enc using $\mathbf{x}_r$ as input, $\mathbf{z}$ is fed into Dec to reconstruct the image $\hat{\mathbf{x}}_{\mathbf{z}}$. Hence, $\mathcal{L}_{L2}$ is defined as:

$$\mathcal{L}_{L2} = \mathbb{E}_{\mathbf{x}_r\sim p_{data}}\Big[\|Dec(Enc(\mathbf{x}_r)) - \mathbf{x}_r\|_2^2\Big] \quad (4)$$

where $\|\cdot\|$ is the second-ordered norm. It should be noted that in the original VAE [8], $\mathcal{L}_{L2}$ is used to update both the Enc and Dec. Conversely, we found that $\mathcal{L}_{L2}$ degrades the quality of the generated images when it is used to update Enc. Hence, we only use $\mathcal{L}_{L2}$ when updating θ_G. The final form of the loss function for G is $\mathcal{L}_G = \mathcal{L}_{adv} + \mathcal{L}_{L2}$.

3. EXPERIMENTS

3.1. Dataset

In this work, we used the publicly available Wikiart dataset [12, 16] for our experiments. Wikiart is the largest public available dataset that contains around 80,000 annotated artwork in terms of genre, artist and style class. However, not all the artwork are annotated in the 3 respective classes. To be specific, all artwork are annotated for the *style* class. But, there are only 60,000 artwork annotated for the *genre* class, and only around 20,000 artwork are annotated for the *artist* class. We split the Wikiart dataset[1] into two parts: 30% for testing and the rest for training.

(a) Real (b) Synthesis

Fig. 3: Sample of the generated *artist* artwork - Gustave Dore (top) and Vincent van Gogh (bottom).

3.2. Experiment Settings

In terms of the ARTGAN architectures, we used $\alpha = 0.2$ for all leaky ReLU. On the other hand, Dec shares the layers Deconv3 to Deconv6 in G; and Enc shares the layers Conv1 to Conv4 in D. We trained the proposed ARTGAN and other models in the experiments for 100 epochs with minibatch size of 128. For stability, we used the adaptive learning method RmsProp [18] for optimization. We set the decay rate to 0.9 and initial learning rate to 0.001. We found out that reducing the learning rate during the training process will help in improving the image quality. Hence, the learning rate is reduced by a factor of 10 at epoch 80.

3.3. Artwork Synthesis Quality

Genre: We compare quality of the generated artwork trained based on the *genre*. Fig. 1 shows samples of the artwork synthetically generated by ARTGAN, DCGAN [11] and GAN/VAE, respectively. We can visually notice that the generated artwork from the DCGAN are relatively poor, with a lot of noises (artefacts) in them. In GAN/VAE, we could notice that the generated artwork are less noisy and look slightly more natural. However, we can observe that they are not as compelling. In contrast, the generated artwork from the proposed ARTGAN are a lot more natural visually in overall.

Artist: Fig. 3 illustrates artwork created by ARTGAN based on *artist* and interestingly, the ARTGAN is able to recognize the artist's preferences. For instance, most of the *Gustave Dore's* masterpieces are completed using engraving, which are usually dull in color as in Fig. 3a-top. Such pattern was captured and led the ARTGAN to draw greyish images as

[1] https://github.com/cs-chan/ICIP2016-PC/

(a) Real (b) Synthesis

Fig. 4: Sample of the generated *style* artwork - Ukiyo-e.

Fig. 5: Generated CIFAR-10 images using DCGAN [11] (top) and ARTGAN (bottom).

depicted in Figure 3b-(top). Similarly, most of the Vincent van Gogh's masterpieces in the Wikiart dataset are annotated as *Sketch and Study* genre as illustrated in Fig. 3a-bottom. In this genre, Van Gogh's palette consisted mainly of sombre earth tones, particularly dark brown, and showed no sign of the vivid colours that distinguish his later work, e.g the famous ***The Starry Night*** masterpiece. This explains why the artwork synthetically generated by ARTGAN is colourless (Fig. 3b-bottom).

Style: Fig. 4 presents the artwork synthetically generated by ARTGAN based on *style*. One interesting observation can be seen on the *Ukiyo-e* style paintings. Generally, this painting style is produced using the woodblock printing for mass production and a large portion of these paintings appear to be yellowish as shown in Figure 4a due to the paper material. Such characteristic can be seen in the generated *Ukiyo-e* style paintings. Although the subjects in the paintings are hardly recognizable, it is noticeable that ARTGAN is trying to mimic the pattern of the subjects.

3.4. Drawing CIFAR-10 with ARTGAN

We trained both the DCGAN [11] and ARTGAN to generate natural images using the CIFAR-10 dataset. The generated samples on CIFAR-10 are presented in Fig. 5. As aforementioned, the DCGAN is able to generate much recognizable images, contrast to its failure in generating artwork. This implies our earlier statements that the objects in CIFAR-10 have a fairly structured shape, and so it is much easier to learn compared to the artwork that are abstract. Even so, we could still

Fig. 6: Nearest neighbour comparisons. Paintings in the red dotted boxes are the corresponding nearest paintings.

Table 1: Comparison between different GAN models using the log-likelihood measured by Parzen window estimate.

Model	Log-likelihood
DCGAN [11]	2348 ± 67
GAE/VAE	2483 ± 67
ARTGAN	$\mathbf{2564 \pm 67}$

notice some of the generated shapes are not as compelling due to CIFAR-10 exhibits huge variability in shapes compared to CUB-200 dataset of birds and LFW dataset of face. Meanwhile, we observe that the proposed ARTGAN is able to generate much better images. For instance, we can see the automobile and horse with clear shape.

3.5. Quantitative Analysis

By using the GAN models trained previously, we measure the log-likelihood of the generated artwork. Following Goodfellow et al. [5], we measure the log-likelihood using the Parzen window estimate. The results are reported in Table 1 and show that the proposed ARTGAN performs the best among the compared models. However, this measurement might be misleading as stated in [17]. In addition, we also find the nearest training examples of the generated artwork by using exhaustive search on L2 norm in the pixel space. The comparisons are visualized in Fig. 6 and it shows that the proposed ARTGAN does not simply memorize the training set.

4. CONCLUSIONS

In this work, we proposed a novel ARTGAN to synthesize much challenging and complex images. In the empirical experiments, we showed that the feedback from the label information during the back-propagation step improves the quality of the generated artwork. A natural extension to this work is to use a deeper ARTGAN to encode more detail concepts. Furthermore, we are also interested in jointly learn these modes, so that ARTGAN can create artwork based on the combination of several modes.

Acknowledgment

This work is supported by the FRGS-MoHE Grant FP004-2016, and the Titan X was donated by NVIDIA.

5. REFERENCES

[1] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern recognition*, 37(9):1757–1771, 2004.

[2] A. Brock, T. Lim, J. Ritchie, and N. Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016.

[3] E. Denton, S. Chintala, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In *NIPS*, pages 1486–1494, 2015.

[4] P. Foxton. How to practise drawing effectly, 2011.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[6] G. B. Huang and E. Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[7] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. 2009.

[8] A. Larsen, K. Sønderby, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

[9] Y. LeCun, C. Cortes, and C. J. Burges. The mnist database of handwritten digits, 1998.

[10] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[11] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

[12] B. Saleh and A. Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *arXiv preprint arXiv:1505.00855*, 2015.

[13] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016.

[14] W. J. Scheirer, L. P. Jain, and T. E. Boult. Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:2317–2324, 2014.

[15] J. T. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390*, 2015.

[16] 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 *ICIP*, pages 3703–3707, 2016.

[17] L. Theis, A. v. d. Oord, and M. Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

[18] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2), 2012.

[19] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.