2018 42nd IEEE International Conference on Computer Software & Applications

Generation of Character Illustrations from Stick Figures using a Modification of Generative Adversarial Network

Yuuya Fukumoto
Graduate School of Bionics, Computer and Media Science
Tokyo University of Technology
Hachioji, Japan
g211702420@edu.teu.ac.jp

Daiki Shimizu
School of Computer Science
Tokyo University of Technology
Hachioji, Japan
c0114271dc@edu.teu.ac.jp

Chihiro Shibata
Graduate School of Bionics, Computer and Media Science
Tokyo University of Technology
Hachioji, Japan
shibatachh@stf.teu.ac.jp

***Abstract*—We propose a modification of generative adversarial networks (GANs) that generate illustrations of human figures from given poses represented by stick figures. In recent years, while various methods that generate images of characters using GANs have been proposed, it is not yet possible for users to freely designate poses of human figures. When generating an image of a character, the pose of the character takes is an important component of its composition. Thus it is necessary fora user who wants to create an illustration to be able to specify the pose easily. We collected a set of illustrations of human figures from the internet, and for each illustration, a simple line drawing that specifies the pose was drawn manually. We constructed a GAN that takes a line drawing as its input and creates an illustration of a person in a pose that matches the line drawing. These networks are learned using the data set we prepared. In this paper, we propose a new network architecture. After constructing two networks both of which have almost the same structure as pix2pix, which is a variant model of GANs, we stack up those networks based on the idea of stack GAN. The experimental results show that, from stick figures representing common poses such as a standing pose, our methods was able to successfully generate images of characters. However, in the case of stick figures having rare poses that were not in the dataset, such as figures raising a hand or lying down, the generated images were blurred and not of a high-quality but still had the desired shapes. By expanding the dataset to include various poses, it is possible to generate diverse poses more precisely.**

***Index Terms*—Deep Learning, Generative Adversarial Networks, Image Generation**

I. INTRODUCTION

Currently, illustrations of characters are used in various applications such as blogs, social network service icons, corporate image characters and local characters for regional revitalization. However, in order for people to draw high-quality illustrations, varied expert knowledge and experience is required such as knowledge of design, anatomy, lighting and shadows for mastering which a large amount of exercise is necessary.

In recent years, with the development of machine-learning technology, creative activities performed by artificial intelligence have been realized. For instance, a project to create a new painting by Rembrandt using deep learning called The Next Rembrandt[1], which consists of data experts and art historians, is one such creative activity.

The objective of our research is to develop an AI system that automatically generates images of characters and can be easily and freely tuned by users using deep learning algorithms. In this paper, we propose the use of variants of generative adversarial networks (GANs) that generate an illustration of a human figure from a manually drawn pose comprising simple lines.

II. RELATED WORK

A. Generative Adversarial Networks

GANs [1], the use of which was proposed by Goodfellow et al., is a generative model using neural networks. It is comprises two separated networks called the *generator* and *discriminator*. A generator takes a stochastic noise vector as its input, and then outputs a fake image that is similar to a real image selected from the training dataset. The discriminator judges whether the input image is fake or real. The generator and the discriminator are trained alternatively such that one defeats the other. Recently, some variants of GANs have been proposed in the research area of image generation.

B. Stack GAN

Stack GAN is a modification of GAN proposed by Zhang et al. [2]. Stack GAN is comprises two stages of GANs.The aim of Stack GAN is to generate fine images from simple texts that describes some image. The generator in Stage I takes a noise vector z and a text and outputs a low-resolution image in which the color and shape of the drawn object are roughly consistent with the text. The discriminator in Stage I judges whether the input image is the output by the generator or a real image associated with the text as a pair in the training dataset. In Stage II, the generator–which is different from that in Stage I–takes the image generated in Stage I and the text

[1]Rembrandt : https://www.nextrembrandt.com

0730-3157/18/$31.00 ©2018 IEEE
DOI 10.1109/COMPSAC.2018.10225

again as an input. It then generates a high-resolution image that is finely modified based on the input image. The discriminator in Stage II again classifies the input image as a generated or a real image.

C. Generation of Character Faces using GANs

Several researches have focused on the image generation of character faces using GANs. In the first such attempt, deep convolutional GAN(DCGAN) [3] was applied to generate images of character's faces [2]. This model generates various faces from stochastic noise vectors without any conditional input. Features of characters such as colors of hairs are automatically clustered in the vector space of the input. Another research proposed the handling of the types of generated images in a supervised manner by inputting some vectors representing the features of the characters to the generator.

It is empirically known that the generator of DCGAN often learns to output distorted images as a DCGAN model is not stable during learning. Jin et al. proposed a new gradient penalty scheme to improve the stability of learning of GAN, called DRAGAN [4]. The image generation of character faces using DRAGAN with supervised features of characters [5] can generate desirable fine images with little distortion.

III. Proposed method

As discussed in the previous section, generating illustrations of characters with a certain level of quality has become possible. The handling of features, such as hair color, has also been studied by several researchers. However, it is not yet possible to generate images of characters whose poses are freely designated. To represent a desired pose quickly, we propose the use of the drawing of a stick figure as the input for the generators of the GANs. We also propose a new architecture of GANs for realizing superior performance in generating illustrations of human figures from given poses represented by stick-figure images.

A. Overview of the Proposed Architecture of GANs

In Stack GAN, text and noise are used as input, and images are generated. In this paper, we propose a GAN model that uses line drawings as its input and outputs as image. Using this model, it is possible to generate more detailed images than other sophisticated GANs such as pix2pix [6].

B. Pix2pix-based Architecture

As a building block for the proposed architecture, an architecture based on pix2pix is used. We describe it in details as follows. The generator is comprises two networks called encoder and decoder.

Each convolutional layer in the encoder network has a corresponding convolutional layer in the decoder network. The corresponding layers in the encoder and decoder output tensors of the same height, width, and number of batches. The output of each layer in the encoder is concatenated with the output of the corresponding layer in the decoder. The concatenated tenser is fed to the upper layer in the decoder. As this concatenation can be thought of as skipping the intermediate layers, it is called the *skip connection*. There are seven convolutional layers in the encoder (or the decoder). The output of enc_i and dec_i is fed to enc_{i+1} and dec_{i-1} respectively. When each convolutional layer is passed through, the height and width of the image are halved in the encoder (or doubled in the decoder). The output of enc_i of the encoder is concatenated with the output of dec_{i+1} and fed to dec_i of the decoder.

The discriminator takes two images as its input. One image is that of the stick figure, which is the same as the input for the encoder. The other iamge is either image generated by the decoder or a true image paired with the image of the stick figure. The former is fed to dis_{0a} and the latter to dis_{0b}. The outputs of dis_{0a} and dis_{0b} are concatenated and fed to dis_1. The output of dis_4 is fed to a fully connected layer the output of which is a real number in the range $[0, 1]$ that represents the probability that the input is a true image for the stick figure.

C. Proposed Architecture

The proposed architecture is based on the previously mentioned notion of stack GAN. Two Pix2pix-like architectures are blocked in the proposed architecture. We call these architectures Stages I and II. The details of the layers in the encoders, decoders, and discriminators are slightly different from the the Pix2pix-based ones described in the previous section. As can be obserbed, the last convolutional layer (dec_1) is missing in Stage 1 and thus the generator in Stage 1 outputs a halved image (64×64). Fig. 1 shows the overview of the proposed architecture in which the pix2pix-based blocks are modified such that they can connect with each other smoothly and generate images well.

IV. Methodology

A. Dataset

To construct the dataset, our script collected illustrations of human figures from the "pose" category on a particular web site[3] containing free-to-use images. After eliminating images that do not represent a human figure, 690 images were stored in the dataset. The images of stick figures representing human figures, which consist of extremely simple line drawings, are manually drawn using a painting tool. As the total number of data is relatively small, the dataset was expanded to 13,800 images by flipping right and left, cropping images randomly to 128×128 and changing the contrast. We used 90% of the dataset as the training data and 10% as the test data.

B. Implementation

The optimization algorithm used in the experiments is Adam. The networks in Stage I and Stage II are learned separately. Stage II is learned after completing the learning of Stage I. Each network is trained for 400 epochs. We implemented both the pix2pix-based and the proposed architecture

[2]https://github.com/mattya/chainer-DCGAN

[3]Irasutoya, https://www.irasutoya.com/

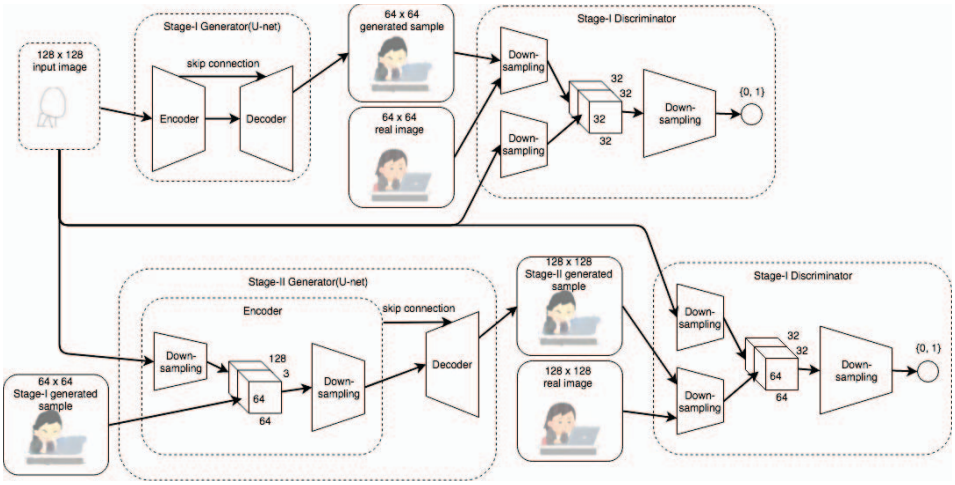


Fig. 1. Overview of the proposed architecture

by modifying an existing implementation for the original pix2pix model[4].

V. Experimental results

A. Experiments for the Pix2pix-based Architecture

First we demonstrate the learning of the pix2pix-based architecture described in Section III-B for a comparison with the proposed architecture described later. The loss for the discriminator represents the average of the $[0, 1]$ losses for both the real and generated images (thus, its range is $[0, 2]$). The loss for the generator represents a weighted sum of the $[0, 1]$ loss for the generated image and the mean absolute error (MAE) between the true and generated images. The latter term is added in order to speed up the learning by forcing generated images to be different from the true image from the beginning. Consequently, the loss value for the generator mostly represents the MAE if that value is much larger than 1.0.

In the earlier stage, the loss for the generator quickly drops to approximately 7.5. This implies that the generator is able to generate images that are similar to the real images in terms of MAE. However, after that, the green line gradually goes up and then down. Finally, the loss ends at a value approximately 7.5, which is almost the as the value at around the 40th epoch. Furthermore, as represented by the blue line, the loss for the discriminator gradually becomes very low until approximately the 200th epoch. After that, although it increases slightly, it remains low. This low loss indicates that the discriminator distinguishes the generated images from the real images fairly accurately. These facts indicate that the losses of the generated images are still dominated by the MAE instead of the discriminator's loss. Consequently, the generated images are not affected so much by the adversarial framework of GANs.

[4]https://github.com/pfnet-research/chainer-pix2pix

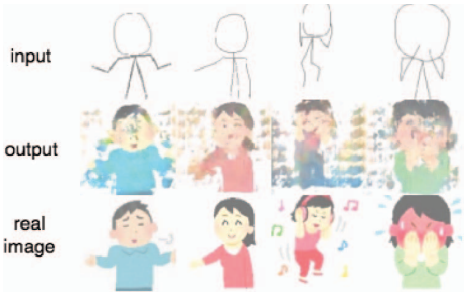


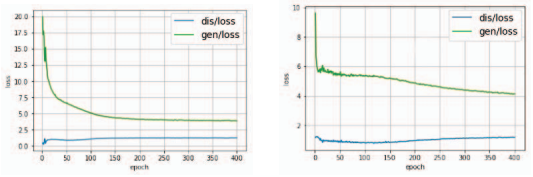
Fig. 2. Results of image generations with the pix2pix-based architecture at the last epoch.

Images are collapsed at the 400th epoch as is obserbed in the bottom left images in Fig. 2. As they are drawn neatly at 380th epoch, it is implied that the generated images are not stable even after several iterations. Even if the pix2pix-based architecture can successfully generate rough images of human figures the poses of which are as intended, the details of the images such as facial parts are blurred and distorted.

B. Experiments for the Proposed Architecture

Fig. 3(a) and Fig. 3(b) show the losses of the generators and discriminators in Stage I and II as functions of the number of epochs. As indicate by the green lines, the losses for the generators steadily drop to approximately 4.0 in both Stage I and Stage II. Furthermore, as indicated by the blue line, the loss for the discriminator gradually increases to 1.0 in Stage I, and to a value higher than 1.0 in Stager II. This means that the generated images of Stage II are more difficult for the discriminator to distinguish from the real images the discriminator than those of Stage I. This indicates that the generated images of Stage II are more similar to the real images for the convolutional neural network called the discriminator. It is important that the generator can finally mimic the discriminator in construct to the pix2pix-based

architecture. We can say that the adversarial learning works well in the proposed method.



(a) Losses as functions of the number of epochs for Stage I

(b) Losses as functions of the number of epochs for Stage II.

Fig. 3. Stack GAN losses: "dis"(blue line) and "gen"(green line) represent the losses for the discriminator and generator respectively.

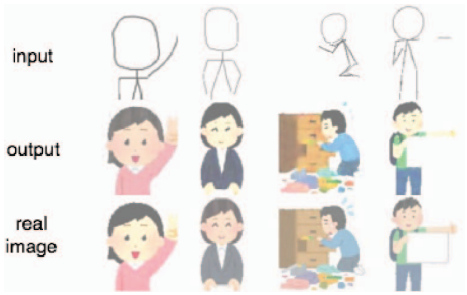


Fig. 4. Results of image generation with the proposed architecture at the last epoch: input images of stick figures **(upper row)**, generated images of the character figures **(middle row)**, and true images of the characters figures **(lower row)**.

The input images, that is stick figures, the generated images, and the real images of Stage II at the last epoch are shown in Fig. 4. As can be obserbed, the details such as the facial parts are generated without blurring in contrast to the pix2pix-based architecture. Moreover, there are some desirable differences between the generated images and the real images: some details such as the face colors are different. Overall, the generated images of the proposed architecture are improved as compared to those of the pix2pix-based architecture.

C. Generation from Rare Poses

The ability to generate details is improved by the proposed architecture as compared with the pix2pix-based architecture. As both the structures of the GANs in Stage 1 and Stage 2 are almost the same as the pix2pix-based model, this improvement is the effect of constructing two-stage GANs. A rough image is generated at the first stage and a fine image was drawn in the second stage. We drew new stick figures that have rare poses such as the upper row of Fig. 5 manually and input the learned generator. The images generated by the models at the the 400th epoch are shown in the middle row and the lower row of Fig. 5, respectively. As shown in Fig. 5, rare poses such those involving bent knees and raised hands could not be successfully generated. There is room for improvement through the use of some contrivances such as making datasets more diverse and learning for a longer period in order to make the adversarial learning more robust.

Fig. 5. Comparison of pix2pix-based and the proposed architectures for rare poses.

VI. Conclusion

In this paper, we proposed a method for generating an illustration of a person from extremely simple line drawings or stick figures. The proposed architecture comprises stacked pix2pix-based blocks. Using the dataset labeled with manually drawn stick-figure images, the proposed GAN was trained. The trained model successfully generates appropriate images from the input stick figures when they represent common poses in the dataset. However, the generated images are blurred and of subpar quality while having desired shapes when stick figures having rare poses–such as figures lying down–that are not in the dataset are input. To solve this problem, it is necessary to expand the dataset such that it comprises diverse poses. As features other than the pose of the character–such as the color of clothes–are not controlled in this model, it is needed to add label information for controlling them. Although we used relatively simple illustrations in the dataset in this study, the proposed model cloud be used to generate more complicated illustrations in practical applications.

Acknowledges

The authors acknowledge the support of the Japan Society for the Promotion of Science (JSPS) KAKENHI grant 26730123.

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[2] Han Zhang, Tao Xu, and Hongsheng Li. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5908–5916, 2017.

[3] Alec Radford, Luke Metz, Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[4] Naveen Kodali, Jacob Abernethy, James Hays, Zsolt Kira. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*, 2017.

[5] Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, and Zhihao Fang. Towards the automatic anime characters creation with generative adversarial networks. *CoRR*, abs/1708.05509, 2017.

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arxiv*, 2016.