Self-supervised GAN for Image Generation by Correlating Image Channels

Sheng Qian[1], Wen-ming Cao[1], Rui Li[1], Si Wu[2], and Hau-san Wong[1(✉)]

[1] Department of Computer Science, City University of Hong Kong, Hong Kong, China
cshswong@cityu.edu.hk

[2] Department of Computer Science, South China University of Technology, Guangzhou, China

Abstract. Current most GAN-based methods directly generate all channels of a color image as a whole, while digging self-supervised information from the correlation between image channels for improving image generation has not been investigated. In this paper, we consider that a color image could be split into multiple sets of channels in terms of channels' semantic, and these sets of channels are closely related rather than completely independent. By leveraging this characteristic of color images, we introduce self-supervised learning into the GAN framework, and propose a generative model called *Self-supervised GAN*. Specifically, we explicitly decompose the generation process as follows: (1) generate image channels, (2) correlate image channels, (3) concatenate image channels into the whole image. Based on these operations, we not only perform a basic adversarial learning task for generating images, but also construct an auxiliary self-supervised learning task for further regularizing generation procedures. Experimental results demonstrate that the proposed method can improve image generation compared with representative methods and possess capabilities of image colorization and image texturization.

Keywords: Image generation · GAN · Self-supervised learning

1 Introduction

Recently increasing attention has been paid to building unsupervised learning models for image generation and representation learning. In general, there are two types of unsupervised learning approaches: (1) a discriminative framework with self-supervised proxy tasks for learning representations; (2) a generative framework for generating data and learning representations [26].

Considering expensive human annotation and plenty of free unlabeled data, self-supervised learning methods directly dig supervised information from the raw data. Based on data characteristics, all of these methods will construct various

© Springer Nature Switzerland AG 2018
R. Hong et al. (Eds.): PCM 2018, LNCS 11165, pp. 78–88, 2018.
https://doi.org/10.1007/978-3-030-00767-6_8

proxy tasks to learn meaningful representations. In computer vision domains both temporal and spatial clues have been proven to be informative signals for constructing proxy tasks, such as egomotion [1], unsupervised object tracking [23], spatial arrangement [7,18], transformations [8], and context-based reconstruction [20]. Besides, the correlation between image channels is also another important clue, such as colorization [3,4,6,13,14,27] and cross-channel prediction [28].

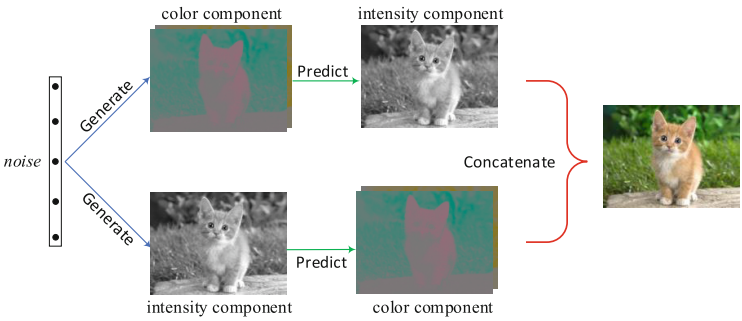


Fig. 1. Image generation by Self-supervised GAN.

Since images are high dimensional with complex patterns, various generative methods have been proposed for achieving better performance of image generation based on the GAN [9] framework. Among them, some methods try to leverage the inherent attributes of images, and focus on improving the architectural design of GAN. For example, [21] exploits the advantages of CNN in image applications, and [5,25,26] design more elaborate network architectures by exploiting structure/style formation [26], multiscale representation [5], and background/foreground composition [25], respectively.

In this paper we expect to incorporate adversarial learning and self-supervised learning into a generative model, and leverage their advantages for improving the performance of image generation. For this purpose, we propose a generative model called *Self-supervised GAN* (*denoted as SSGAN*). Specifically, we exploit one of the most basic characteristics of color images as follows: (1) a color image is composed of multiple channels which can be grouped into specific sets based on channels' semantic; and (2) these sets of channels have a close relationship. To simplify the following discussion, we focus on the case where a color image is generally split into the following two components: intensity and color. Considering the above characteristic of color images, as illustrated in Fig. 1, the generation process can be decomposed into the following procedures to generate the whole image: (a) generate two sets of channels; (b) transform from one set to the other set; (c) concatenate these two sets to form the whole data.

Based on these operations, we could combine adversarial learning and self-supervised learning together. Except for performing the adversarial learning task for image generation, we also construct the self-supervised learning task

where different sets of channels predict each other using true data to further improve generation. Viewed from another perspective, most of the existing methods directly generate all channels of color images as a whole, and only exploit self-supervised information from true/fake data. Compared with these methods, our proposed method could further dig more self-supervised information from the correlation between image channels. Overall, the main contributions of this work are as follows:

- By leveraging the relationship between color image channels, we propose a generative model which can well incorporate adversarial learning and self-supervised learning and improve the performance of image generation.
- Except for performing image generation, the proposed model also possesses capabilities of image colorization and image texturization.

In the experiments we conduct both qualitative and quantitative evaluation on the benchmark dataset, and compare the proposed method with several representative methods. The experimental results verify the effectiveness of our method.

2 Related Work

2.1 Adversarial Learning

Generally GAN-based methods focus on improving two factors of GAN: the architectural design and the train criteria, since these factors have a great influence on the performance of image generation. For the architectural design, [21] propose to stabilize GAN by applying architecture guidelines of CNN. By further exploiting the inherent attributes of images, [5,26] cascade multiple GANs and adopt a multi-scale strategy, and [24,25] analyze the image formation and decompose image generation into cascaded procedures. Besides, [11,15] design symmetrical architectures to model the cross-domain relationship of two image domains by coupling two GANs in parallel and in cross-linked respectively. For the train criteria, [16] adopts the least squares loss instead of the cross entropy loss used by GAN, and [19] further extends GAN in the f-divergences estimation framework. Differently, [29] rephrases the adversarial learning of GAN from the perspective of an energy-based model. Besides, [2,10] propose to measure the distribution discrepancy using Earth-Mover distance. Instead of weight clipping using by [2,10,17] penalizes the norm of the discriminator's gradient for enforcing a Lipschitz constraint. Overall these GAN-based methods can improve the training stability of models and the performance of image generation.

2.2 Self-supervised Learning

All of the self-supervised methods will leverage discriminative proxy tasks to learn representations well transferred to downstream tasks. By learning representations invariant to transformations, [1] predicts the transformation between

a pair of adjacent frames, [23] considers a pair of identically tracked patches from successive frames to make their distance in the latent representation space more closer, and [8] generically forms a set of surrogate classes by applying vast image transformations to images. Considering the spatial arrangement of image patches, [7] predicts the relative position of two image patches, [18] solves the jigsaw puzzle composed of a set of object's patches, and [20] proposes the context-encoder to reconstruct the image region from its contextual region with an adversarial regularization. Some works focus on the problem of image colorization based on the regression model [4,6] or the classification model [13,14,27]. Furthermore, [3] improves the image diversity of colorization via leveraging conditional adversarial learning, and [28] proposes a split-brain auto-encoder by splitting the whole image into multiple channels and performing cross-channel prediction tasks.

3 Preliminary for Adversarial Learning

The GAN framework is an approach for estimating generative models via an adversarial learning process. Specifically, its network architecture is composed of a generator G and a discriminator D. Its objective is to make D to correctly differentiate between the true data and the generated data, and propel G to well capture the data distribution. Considering the training difficulty of the original GAN, we use SNGAN [17] as the baseline model since it shows better generation performance and training stabilization. Formally, the value function and the spectral normalization term adopted by SNGAN are as follows:

$$\begin{aligned} &L_{gan} = \mathbb{E}_{x\sim p_x(x)}[log(D(x))] + \mathbb{E}_{z\sim p_z(z)}[log(1 - D(G(z)))], \\ &SN(W^l) := W^l/\sigma(W^l) \quad where \quad W^l \in \theta, \end{aligned} \tag{1}$$

where $p_x(x)$ and $p_z(z)$ are the true data distribution and the prior noise distribution, respectively. $\theta := \{W^1, ..., W^n\}$ is the parameter set of the discriminator's layers, n is the number of layers, and $\sigma(\cdot)$ is the spectral norm of a matrix. More details about the spectral normalization can refer to [17].

4 Self-supervised GAN

In this section we introduce the proposed generative model in detail, and focus on the following aspects: network architecture, adversarial learning for image generation, self-supervised learning for generation regularization, and model training.

4.1 Network Architecture

To perform the basic adversarial learning task and the auxiliary self-supervised learning task, we design an elaborate network architecture as shown in Fig. 2. Specifically, this architecture consists of two types of components for generation

and discrimination, and all components are parameterized by deep neural networks. Among them, $S_1 \circ G$ and $S_2 \circ G$ are generators for two sets of channels, where G is the shared part for both sets, and S_1 and S_2 are the splitting parts for each set. Since there are two types of cross-channel prediction: (1) predicting the color component from the intensity component; (2) predicting the intensity component from the color component, we design two transformers T_{12} and T_{21} for predicting one set from the other set. C is a concatenator for combining two sets to form the whole data. D_1, D_2 and D_x are discriminators for the first set of channels, the second set of channels and the whole data, respectively.

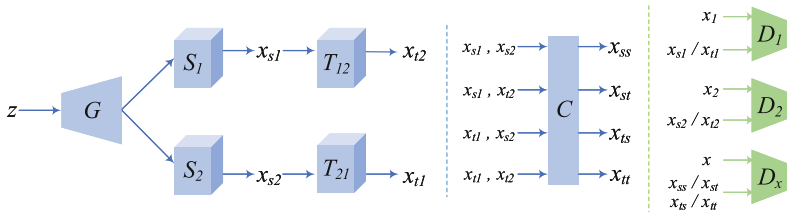


Fig. 2. The network architecture of SSGAN.

4.2 Adversarial Learning for Image Generation

As shown in Fig. 2, given a noise sample $z \sim p_z(z)$ we can generate two splitting channels (x_{s1} and x_{s2}) and two transformed channels (x_{t2} and x_{t1}), and concatenate these channels into four types of the whole data (x_{ss}, x_{st}, x_{ts} and x_{tt}). Overall, they are given by

$$\begin{aligned} &x_{s1} = S_1 \circ G(z),\ x_{s2} = S_2 \circ G(z),\ x_{t2} = T_{12}(x_{s1}),\ x_{t1} = T_{21}(x_{s2}); \\ &x_{ss} = C(x_{s1}, x_{s2}),\ x_{st} = C(x_{s1}, x_{t2}),\ x_{ts} = C(x_{t1}, x_{s2}),\ x_{tt} = C(x_{t1}, x_{t2}). \end{aligned} \tag{2}$$

By generating and concatenating image channels, we can build three types of generative models — GM_1, GM_2 and GM_x, as shown in Table 1. These models are responsible for the following adversarial learning tasks respectively: learning the distributions of (1) the first set of channels, (2) the second set of channels, and (3) the whole data. Following SNGAN, the corresponding value functions of these models are as follows:

$$\begin{aligned} L_1 &= \mathbb{E}[log(D_1(x_1))] + \mathbb{E}[log(1 - D_1(x_{*1}))], \\ L_2 &= \mathbb{E}[log(D_2(x_2))] + \mathbb{E}[log(1 - D_2(x_{*2}))], \\ L_x &= \mathbb{E}[log(D_x(x))] + \mathbb{E}[log(1 - D_x(x_{**}))], \end{aligned} \tag{3}$$

where x_{*1} and x_{*2} denote the generated channels; x_{**} denotes the concatenated whole data; x_1 and x_2 are two sets of channels from the true whole data x. For simplicity, the spectral normalization term of each model is ignored here.

Table 1. Three types of generative models.

Generative model	Components	
	Generation	Discrimination
GM_1	$S_1 \circ G$; $T_{21} \circ S_2 \circ G$	D_1
GM_2	$S_2 \circ G$; $T_{12} \circ S_1 \circ G$	D_2
GM_x	$S_1 \circ G$; $S_2 \circ G$; $T_{12} \circ S_1 \circ G$; $T_{21} \circ S_2 \circ G$	D_x

4.3 Self-supervised Learning for Generation Regularization

Except for adversarial learning for image generation, we further introduce a self-supervised learning task to improve image generation. This task performs a cross-channel prediction by only exploiting true data. Specifically, we split the true data x into x_1 and x_2, reuse transformers T_{12} and T_{21} as cross-channel predictors, and generate two predicting sets of channels — $T_{12}(x_1)$ and $T_{21}(x_2)$. The corresponding loss functions of cross-channel predictors are as follows:

$$L_{T_{12}} = \mathbb{E}[\ell(T_{12}(x_1), x_2)] \quad and \quad L_{T_{21}} = \mathbb{E}[\ell(T_{21}(x_2), x_1)], \tag{4}$$

where $\ell(m, n) = \|m - n\|_p$ measures the reconstruction error of two image channels based on the $\mathbf{L}^p$ norm, and we set $\mathbf{L}^1$ in this paper.

4.4 Model Training

Considering the proposed network architecture and two types of learning tasks, we can train the proposed model in two stages: (1) train these components ($S_1 \circ G$, $S_2 \circ G$ for generation; D_1, D_2, D_x for discrimination) and transformers (T_{12}, T_{21}) independently; and (2) train all components jointly. When jointly training all components, it should be noted that some components are affected by multiple value functions. Hence, we should balance the above value functions.

5 Experiments

We evaluate the proposed SSGAN on the benchmark dataset CIFAR [12], and provide both quantitative and qualitative evaluation. Specifically, we focus on the following aspects: image generation, inspecting the effect of self-supervised learning and channel prediction. For quantitative evaluation of the generation performance, we adopt the inception score (denoted as *IS*) [22]. We choose the *RGB* and *Lab* color spaces, where the *RGB* color space is used for the baselines and the *Lab* color space is used for the SSGAN. Briefly speaking, a whole *Lab* image could be divided into the intensity channel L and the color channels ab in the SSGAN.

Besides, some key configurations of experimental implementation are listed as follows. (1) *Network architecture:* we follow the CNN architectures [17].

(2) *Optimizer:* we use Adam optimizer for optimization with learning rate ($\alpha = 0.0001$) and the first and second order momentum parameters ($\beta_1 = 0.5$ and $\beta_2 = 0.999$) [17]. (3) *Model Training:* to balance the above value functions, we set the coefficient of L_{T_*} as 10 by experience, so that both the adversarial learning task and the self-supervised learning task can contribute to model learning.

5.1 Image Generation

In the SSGAN we can generate four types of the whole image — x_{ss}, x_{st}, x_{ts} and x_{tt}. To compare their performance of image generation, we show four types of generated image samples and list their ISs. In Fig. 3 we can observe that there is not obvious difference between image samples of x_{ss} and x_{st} in terms of visual perception, but image samples of x_{ts} and x_{tt} are inferior than those of x_{ss} and x_{st} in terms of texture and detail (*for a better view by zooming in*). Further, from Table 2 we can see that the IS of x_{st} is the highest, and the ISs of x_{ss} and x_{st} are higher than those of x_{ts} and x_{tt}. Both results indicate that the first type of cross-channel prediction is beneficial to image generation, however the second type of cross-channel prediction does not have a positive effect on image generation.

(a) x_{ss} (b) x_{st} (c) x_{ts} (d) x_{tt}

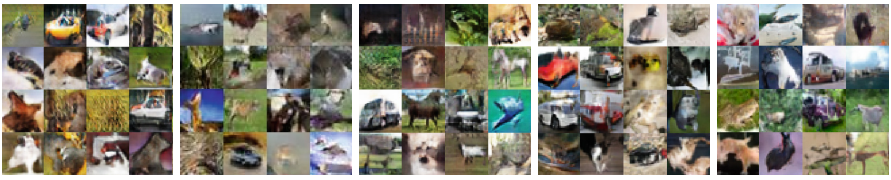
Fig. 3. Four types of image samples generated on CIFAR.

Table 2. Inception scores of four types of the whole image.

Type	Concatenation of (* intensity, * color)	Inception score
x_{ss}	(*splitting*, *splitting*)	7.44 ± 0.09
x_{st}	(*splitting*, *transformed*)	$\mathbf{7.70 \pm 0.09}$
x_{ts}	(*transformed*, *splitting*)	6.86 ± 0.07
x_{tt}	(*transformed*, *transformed*)	6.41 ± 0.08

To compare SSGAN with other methods, we also show image samples generated by these methods and list their ISs. In Fig. 4 images generated by SNGAN and SSGAN are clearer than those by other methods, while there is not obvious

difference between SNGAN and SSGAN in terms of visual perception. However, from Table 3 we can see that the IS of SSGAN improves almost 0.28 compared with the baseline SNGAN. Besides, SSGAN performs better than other methods which directly generate *RGB* images as a whole.



(a) DCGAN (b) WGAN (c) WGAN-GP (d) SNGAN (e) **SSGAN**

Fig. 4. Image samples generated by contrast methods and SSGAN on CIFAR.

Table 3. Inception scores of several representative methods and SSGAN.

Method	Inception score
Real Images	11.24 ± 0.12
DCGAN	6.16 ± 0.17
WGAN	6.41 ± 0.11
WGAN-GP	6.68 ± 0.06
SNGAN	7.42 ± 0.08
SSGAN	$\mathbf{7.70 \pm 0.09}$

5.2 Effect of Self-supervised Learning

In order to evaluate the effectiveness of introducing self-supervised learning, we perform the experiment in which the self-supervised learning for transformer regularization is ignored. In other words, $L_{T_{12}}$ and $L_{T_{21}}$ will be not used for model updating. Here we mainly consider the generated whole image x_{st} and the first type of cross-channel prediction as described in Sect. 4.1. We present the ISs of x_{st} with/without self-supervised learning, and show image samples which consist of the original images and their reconstructed images based on cross-channel prediction. From Table 4 we can see that the IS of x_{st} with self-supervised learning is higher than that of x_{st} without self-supervised learning. As shown in Fig. 5 reconstructed images without self-supervised learning (*the left pair*) fail to infer the color component from the intensity component, while reconstructed images with self-supervised learning (*the right pair*) can better predict the color component. These again indicate that the first type of cross-channel prediction is beneficial to image generation.

Table 4. The effect of self-supervised learning.

Self-supervised	Inception score of x_{st}
Without	7.41 ± 0.07
With	$\mathbf{7.70 \pm 0.09}$

Fig. 5. Reconstructions based on predicting the color component from the intensity component. Each pair consists of the original image and its reconstruction.

5.3 Cross-Channel Prediction

Since we introduce a self-supervised learning task which performs cross-channel prediction, we could reconstruct a color image if only its intensity component or its color component is provided. In other words, the transformers T_{12} and T_{21} of SSGAN also can be used for image colorization and image texturization, respectively.

We illustrate some examples of image colorization and image texturization in Fig. 6. Specifically, the left subfigure includes original images, the middle subfigure includes reconstructed images based on predicting the color component from the given intensity component, and the right subfigure includes reconstructed images based on predicting the intensity component from the given color component. So the middle subfigure and the right subfigure correspond to image colorization and image texturization, respectively. By comparing original images with two types of reconstructed images, we can see that the transformer T_{12} can infer realistic colors, while T_{21} can not predict very fine texture. Viewed from another perspective, it indicates that when performing cross-channel prediction task, the second type is more difficult to the first type. This maybe explain the inferior generation performance of x_{ts} and x_{tt}.

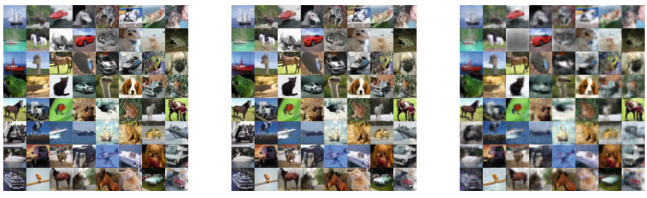


Fig. 6. Reconstructions based on cross-channel prediction.

6 Conclusion

In this work we propose a generative model called *Self-supervised GAN* for improving image generation by introducing self-supervised learning into the GAN framework. Considering that channels of a color image are tightly correlated, we leverage this inherent attribute of color images and explicitly decompose image generation into multiple procedures. Based on the decomposition of image generation, the correlation between image channels as the self-supervised signal is dug for improving image generation. Hence, except for performing the basic image generation task in the adversarial learning framework, we also build an auxiliary cross-channel prediction task to regularize generation procedures in the self-supervised learning framework. Experimental results demonstrate that the proposed method can improve image generation compared with representative methods, and show capabilities of image colorization and image texturization.

References

1. Agrawal, P., Carreira, J., Malik, J.: Learning to see by moving. In: ICCV, pp. 37–45 (2015)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. CoRR arXiv:1701.07875 (2017)
3. Cao, Y., Zhou, Z., Zhang, W., Yu, Y.: Unsupervised diverse colorization via generative adversarial networks. In: Ceci, M., Hollmén, J., Todorovski, L., Vens, C., Džeroski, S. (eds.) ECML PKDD 2017. LNCS (LNAI), vol. 10534, pp. 151–166. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71249-9_10
4. Cheng, Z., Yang, Q., Sheng, B.: Deep colorization. In: ICCV, pp. 415–423 (2015)
5. Denton, E.L., Chintala, S., Szlam, A., Fergus, R.: Deep generative image models using a laplacian pyramid of adversarial networks. In: NIPS, pp. 1486–1494 (2015)
6. Deshpande, A., Rock, J., Forsyth, D.A.: Learning large-scale automatic image colorization. In: ICCV, pp. 567–575 (2015)
7. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: ICCV. pp. 1422–1430 (2015)
8. Dosovitskiy, A., Fischer, P., Springenberg, J.T., Riedmiller, M.A., Brox, T.: Discriminative unsupervised feature learning with exemplar convolutional neural networks. PAMI **38**(9), 1734–1747 (2016)
9. Goodfellow, I.J., et al.: Generative adversarial nets. In: NIPS, pp. 2672–2680 (2014)
10. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: NIPS, pp. 5769–5779 (2017)
11. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: ICML, pp. 1857–1865 (2017)
12. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
13. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 577–593. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_35

14. Larsson, G., Maire, M., Shakhnarovich, G.: Colorization as a proxy task for visual understanding. In: CVPR, pp. 840–849 (2017)
15. Liu, M., Tuzel, O.: Coupled generative adversarial networks. In: NIPS, pp. 469–477 (2016)
16. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z.: Multi-class generative adversarial networks with the L2 loss function. CoRR arXiv:1611.04076 (2016)
17. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. CoRR arXiv:1802.05957 (2018)
18. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 69–84. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_5
19. Nowozin, S., Cseke, B., Tomioka, R.: F-gan: Training generative neural samplers using variational divergence minimization. In: NIPS, pp. 271–279 (2016)
20. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: feature learning by inpainting. In: CVPR, pp. 2536–2544 (2016)
21. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR arXiv:1511.06434 (2015)
22. Salimans, T., Goodfellow, I.J., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: NIPS, pp. 2226–2234 (2016)
23. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: ICCV, pp. 2794–2802 (2015)
24. Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 318–335. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_20
25. Yang, J., Kannan, A., Batra, D., Parikh, D.: LR-GAN: layered recursive generative adversarial networks for image generation. CoRR arXiv:1703.01560 (2017)
26. Zhang, H., et al.: Stackgan: text to photo-realistic image synthesis with stacked generative adversarial networks. CoRR arXiv:1612.03242 (2016)
27. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV, pp. 649–666 (2016)
28. Zhang, R., Isola, P., Efros, A.A.: Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In: CVPR, pp. 645–654 (2017)
29. Zhao, J.J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. CoRR arXiv:1609.03126 (2016)