StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks

Hongzhi Liu

September 29, 2018

CONTENTS

Introduction

1. Abstract

Generating photo-realistic images from text is an important problem and has tremendous applications, including photo-editing, computer-aided design and etc.

Samples generated by existing text-to-image approaches can roughly reflect the meaning of the given descriptions, but they fail to contain necessary details and vivid object parts.

H. Zhang, et al. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

Introduction

1. Abstract

In this paper, the author proposed Stacked Generative Adversarial Networks (StackGAN) to generate 256 × 256 photo-realistic images (128 × 128 previous) conditioned on text descriptions.

A small yellow bird with a black crown and a short black pointed beak

Figure 1. Image size comparison.

S. Reed, et al. Generative adversarial text-to-image synthesis. In *ICML*, 2016.
S. Reed, et al. Learning what and where to draw. In *NIPS*, 2016.

2. Related Work

GAN: D and G play the following two-player minimax game with value function V(G , D) as Eq. 1.

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]. \quad (1)$$

Conditional GAN: The objective function of a two-player minimax game as Eq. 2.

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x}|\boldsymbol{y})] + \mathbb{E}_{\boldsymbol{z}\sim p_{z}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}|\boldsymbol{y})))]. \quad (2)$$

Mehdi Mirza, et al. Conditional Generative Adversarial Nets. *Computer Science*, 2014.

1. The architecture of StackGAN

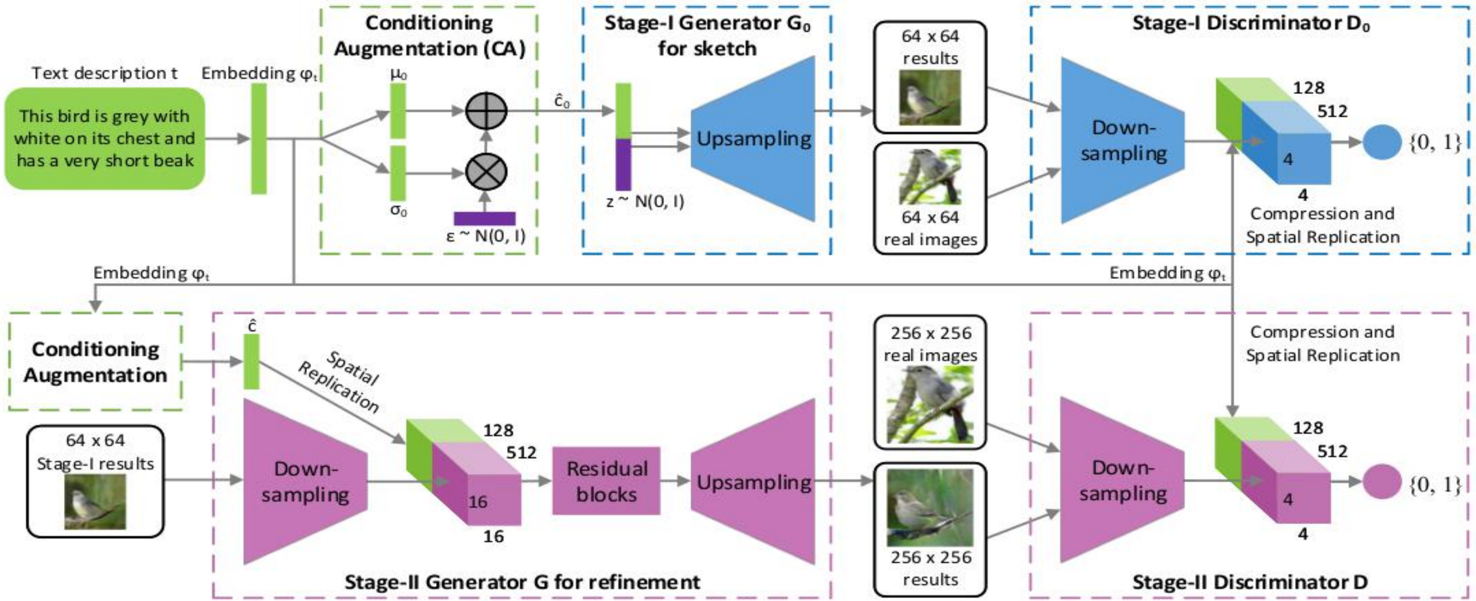

Figure 2: The architecture of the proposed StackGAN. The Stage-I generator draws a low-resolution image by sketching rough shape and basic colors of the object from the given text and painting the background from a random noise vector. Conditioned on Stage-I results, the Stage-II generator corrects defects and adds compelling details into Stage-I results, yielding a more realistic high-resolution image.

2. Method of StackGAN

It decomposes the text-to-image generative process into two stages.

Stage-I GAN: It sketches the primitive shape and basic colors of the object conditioned on the given text description.

Stage-II GAN: It corrects defects in the low-resolution image from Stage-I and completes details of the object by reading the text description again, producing a high-resolution photo-realistic image.

3. Stage-I GAN

The team simplify the task to first generate a low-resolution image with their Stage-I GAN, which focuses on drawing only rough shape and correct colors for the object. Stage-I GAN trains the discriminator D_0 and the generator G_0 by alternatively maximizing L_{D0} in Eq. (3) and minimizing L_{G0} in Eq. (4).

$$\begin{aligned}\mathcal{L}_{D_0} = {} & \mathbb{E}_{(I_0,t)\sim p_{data}}[\log D_0(I_0, \varphi_t)] + \\ & \mathbb{E}_{z\sim p_z, t\sim p_{data}}[\log(1 - D_0(G_0(z, \hat{c}_0), \varphi_t))],\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{L}_{G_0} = {} & \mathbb{E}_{z\sim p_z, t\sim p_{data}}[\log(1 - D_0(G_0(z, \hat{c}_0), \varphi_t))] + \\ & \lambda D_{KL}(\mathcal{N}(\mu_0(\varphi_t), \Sigma_0(\varphi_t)) \,||\, \mathcal{N}(0, I)),\end{aligned} \tag{4}$$

4. Stage-II GAN

The Stage-II GAN is built upon Stage-I GAN results to generate high-resolution images. The Stage-II GAN completes previously ignored text information to generate more photo-realistic details. The D and G in Stage-II GAN are trained by alternatively maximizing L_D in Eq. (5) and minimizing L_G in Eq. (6).

$$\begin{aligned}\mathcal{L}_D = \; & \mathbb{E}_{(I,t)\sim p_{data}}[\log D(I,\varphi_t)] + \\ & \mathbb{E}_{s_0\sim p_{G_0},t\sim p_{data}}[\log(1-D(G(s_0,\hat{c}),\varphi_t))],\end{aligned} \tag{5}$$

$$\begin{aligned}\mathcal{L}_G = \; & \mathbb{E}_{s_0\sim p_{G_0},t\sim p_{data}}[\log(1-D(G(s_0,\hat{c}),\varphi_t))] + \\ & \lambda D_{KL}(\mathcal{N}(\mu(\varphi_t),\Sigma(\varphi_t))\,||\,\mathcal{N}(0,I)),\end{aligned} \tag{6}$$

Experimental Results

1. Generate Pictures

Figure 3. Generate samples from text descriptions from CUB test set.

Experimental Results

2. Performance Comparison

Figure 4. Example results by our StackGAN, GAWWN, and GAN-INT-CLS conditioned on text descriptions from CUB test set.

Experimental Results

3. Inception Scores

Metric	Dataset	GAN-INT-CLS	GAWWN	Our StackGAN
Inception score	CUB	$2.88 \pm .04$	$3.62 \pm .07$	**$3.70 \pm .04$**
	Oxford	$2.66 \pm .03$	/	**$3.20 \pm .01$**
	COCO	$7.88 \pm .07$	/	**$8.45 \pm .03$**
Human rank	CUB	$2.81 \pm .03$	$1.99 \pm .04$	**$1.37 \pm .02$**
	Oxford	$1.87 \pm .03$	/	**$1.13 \pm .03$**
	COCO	$1.89 \pm .04$	/	**$1.11 \pm .03$**

Table 1. Inception scores and average human ranks of our StackGAN, GAWWN, and GAN-INT-CLS on CUB, Oxford-102, and MS-OCO datasets.

Further Research

1. Improve the diversity of the generated samples.

2. Improves the quality of generated images and stabilizes the GANs' training by jointly approximating multiple distributions.

Q & A