[Wang21-WACV] Faces $\grave{a}$ la Carte: Text-to-Face Generation via Attribute Disentanglement

Jiin Kim

1. Introduction

Introduction

- TTI (Text-To-Image)의 목표는 이미지 공간과 텍스트 공간 사이에서 해석 가능한 mapping 을 하는 것이다.
- 자연어는 매우 고차원적인 정보를 가지고 이미지보다 훨씬 추상적인 정보를 가지기 때문에 어려운 과제이다.

Introduction

- 새 또는 꽃 이미지들이 TTI 연구에 흔히 사용되기 때문에, 얼굴 이미지와 텍스트 묘사 사이의 연결은 약하다.
- 그리고 몇 문장의 설명 가지고 모든 인간의 얼굴 특징을 다룰 수는 없다. 또한 같은 얼굴에 대해서 여러 사람이 다른 묘사를 할 수 있다.
- 이렇듯 얼굴 묘사와 얼굴 특징들 사이에서 만족스러운 mappings 는 찾기 어렵다.
- 그러므로, TTF model 은 같은 설명문에 주어질 때 변이가 큰 이미지 그룹을 생성할 수 있어야 한다.

Introduction

- 본 논문은 다음과 같은 novel TTF framework 를 가진 GAN model 을 제안한다.
  - 고품질의 이미지 생성
  - 합성된 이미지와 그에 대한 설명문의 일관성 향상
  - 같은 설명문으로부터 다양한 얼굴 그룹 생성

Introduction

Figure 1: Several examples of synthesised face images produced by our model. We select four groups of images which are arranged according to gender and age. The highlighted features in the text above are from the text annotations provided by the CelebA database [14]. In addition to good rendering accuracy of the specified features, the images show significant variation in terms of unspecified features.

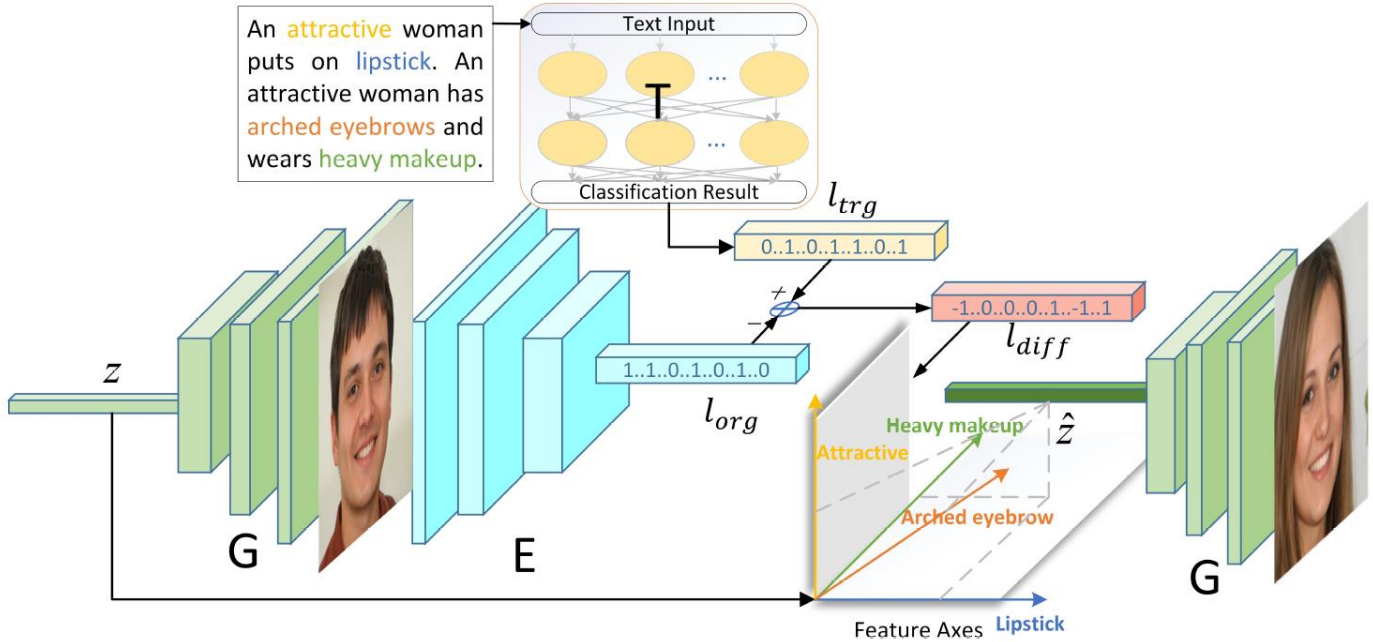
1.2. Contributions

Contributions

1. Image label encoder 인 multi-label text classifier 로 구성된 **TTF-HD** 프레임워크를 제안하고, 다양한 범위의 변이와 함께 매우 질 높은 얼굴을 생성하기 위해 feature-disentangled image generator 를 제안한다.
2. Input noise vector 의 궤적을 안내하기 위해 40개의 label 을 가진 **직교 좌표계**를 추가한다.
3. 조정된 noise vectors 를 disentangled feature space 로 매핑하기 위해 **StyleGAN2** 를 generator 로 사용하여 1024 x 1024 고화질 이미지를 생성한다.

3. Proposed Method

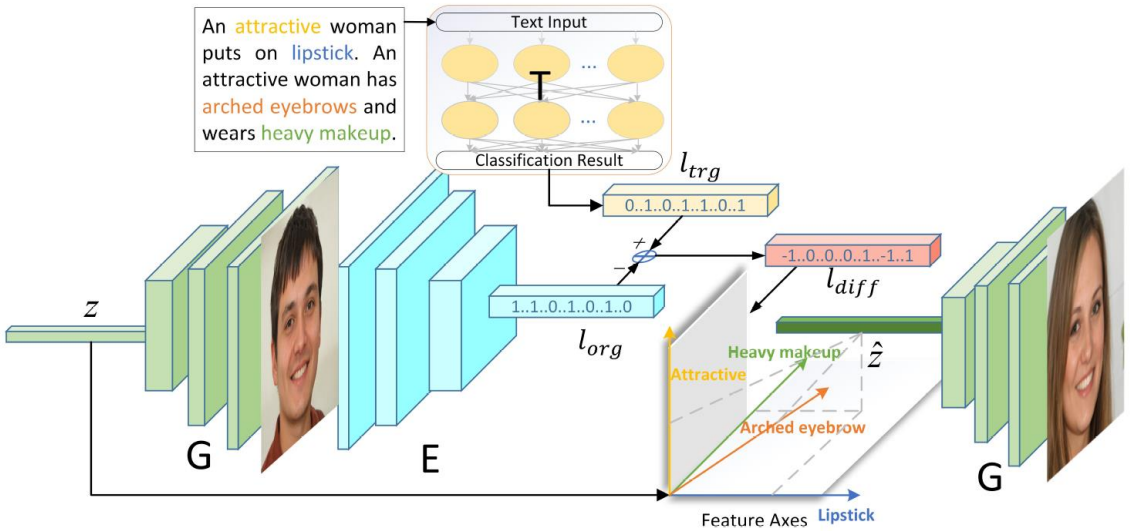
Proposed Method



- TTF-HD 모델의 다이어그램이다.
- Multi-label classifier T, Image encoder E, Generator G 로 구성된다.
- 문장이 multi-label classifier T 에 들어가면 40개의 얼굴 특성을 나타내는 text vector l_{trg} 가 출력된다.
- Image generator G 는 먼저 random noise vector z 로부터 이미지를 합성한다.
- 그런 후, Image encoder E 는 image embeddings l_{org} 를 출력한다.
- Differentiated embedding l_{diff} 는 z 에서 $\hat{z}$ 로 original noise vector 를 조정하는 데 사용된다.

3.1. Multi-Label Text Classification

Multi-Label Text Classification



- TTF 를 수행하기 위해 얼굴 전체를 묘사하는 충분한 facial attribute labels 를 가지는 것은 매우 중요하다.
- 본 논문은 각 얼굴 당 40 개의 facial attribute labels 를 가지는 CelebA dataset 를 사용한다.
- 자유 형식인 설명문을 40개의 facial attributes 에 매핑하기 원한다. 그래서 multi-label text classifier T 를 사용하여 길이가 40인 text embeddings 를 얻는다.
- 설명문의 keywords 는 CelebA dataset 에서 text labels 와 같은 단어 또는 동의어이다.

Multi-Label Text Classification

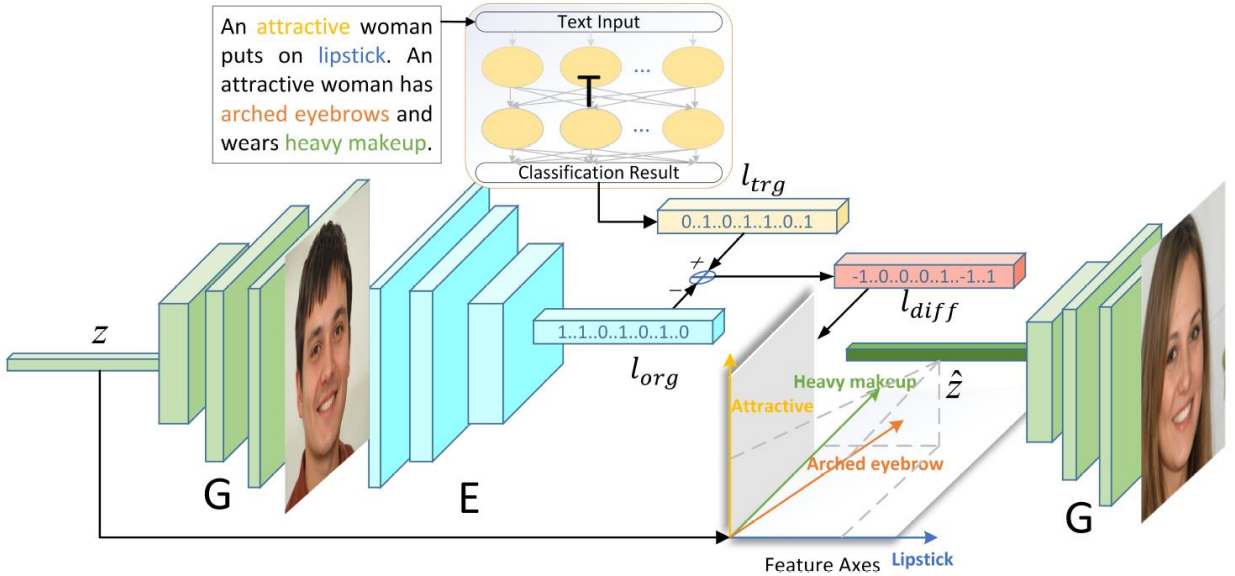
- 본 논문은 최신 자연어 처리 모델인 Bidirectional Transformer(BERT) 를 채택한다.

Figure 3: A possible classification result of the text classifier T.

- CelebA 데이터셋에는 "Yong"만 있고 "Age"는 없다. 그러므로 "young" 는 1과 가까운 수로 나타내고 "old" 는 0과 가까운 수로 나타낸다. 특정하지 않은 특징은 0으로 나타낸다.
- Classifier 는 각 설명문에 길이가 40인 text vector 를 출력한다.
- 이전의 text encoders 와 비교해서 본 논문은 text classifier 는 text descriptions 의 길이에 제한이 없다.

3.2. Image Multi-Label Embeddings

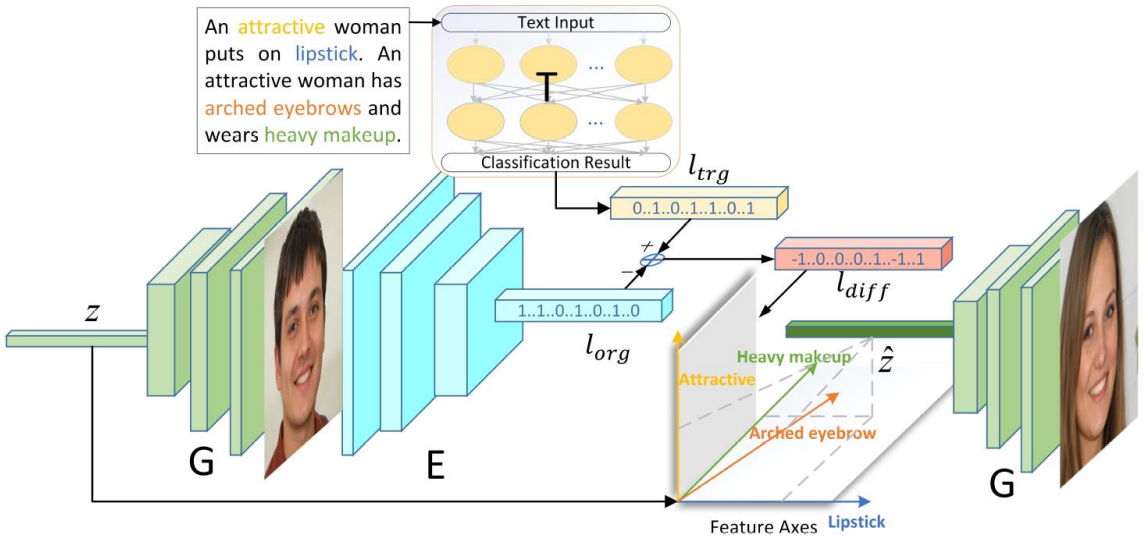
Image Multi-Label Embeddings



- Image encoder E 는 생성된 이미지의 feature labels 를 예측하도록 요구된다.
- 이것을 위해, 본 논문은 MobileNet 모델을 미세하게 조정하여 사용했다.
- MobileNet 을 선택한 이유는 accuracy 와 speed 사이에서 괜찮은 trade-off 를 가진 경량 네트워크 모델이기 때문이다.
- 이 모델을 사용해서 noise vectors 로부터 생성된 이미지의 text vectors 와 같은 길이를 가진 image embeddings 를 얻을 수 있다.

3.3. Feature Axes

Feature Axes



- Image encoder 를 훈련한 후에, 로지스틱 회귀를 사용하여 noise vectors 와 predicted feature labels 사이에서 관계를 찾을 수 있다.
- noise vectors 의 길이는 512이고 feature vector 의 길이는 40이다. 그러므로, 우리는 다음의 식을 얻을 수 있다.
- $y = x \cdot B$
  - B : (512, 40) / x : (1, 512) / y : (1, 40)

Feature Axes

- 모든 attributes 를 분리해야 하기 때문에 직교 좌표계가 필요하다. 직교 좌표계에서 noise vectors 는 특정 feature 축을 움직일 수 있다.
- Gram-Schmidt process 를 통한 projection operator 는 다음과 같다.
  - Gram-Schmidt 는 어떤 벡터 n 개가 주어졌을 때, 이 벡터들을 이용하여 서로 직교하는 정규 벡터로 변환하는 방법이다.

$$\mathrm{proj}_{\mathbf{u}}(\mathbf{v}) = \frac{\langle \mathbf{v}, \mathbf{u} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \mathbf{u} \qquad (2)$$

  - v 는 orthogonalized 된 axis 이고 u 는 reference axis 이다.

$$\mathbf{u}_k = \mathbf{v}_k - \sum_{j=1}^{k-1} \mathrm{proj}_{\mathbf{u}_j}(\mathbf{v}_k), \qquad (3)$$

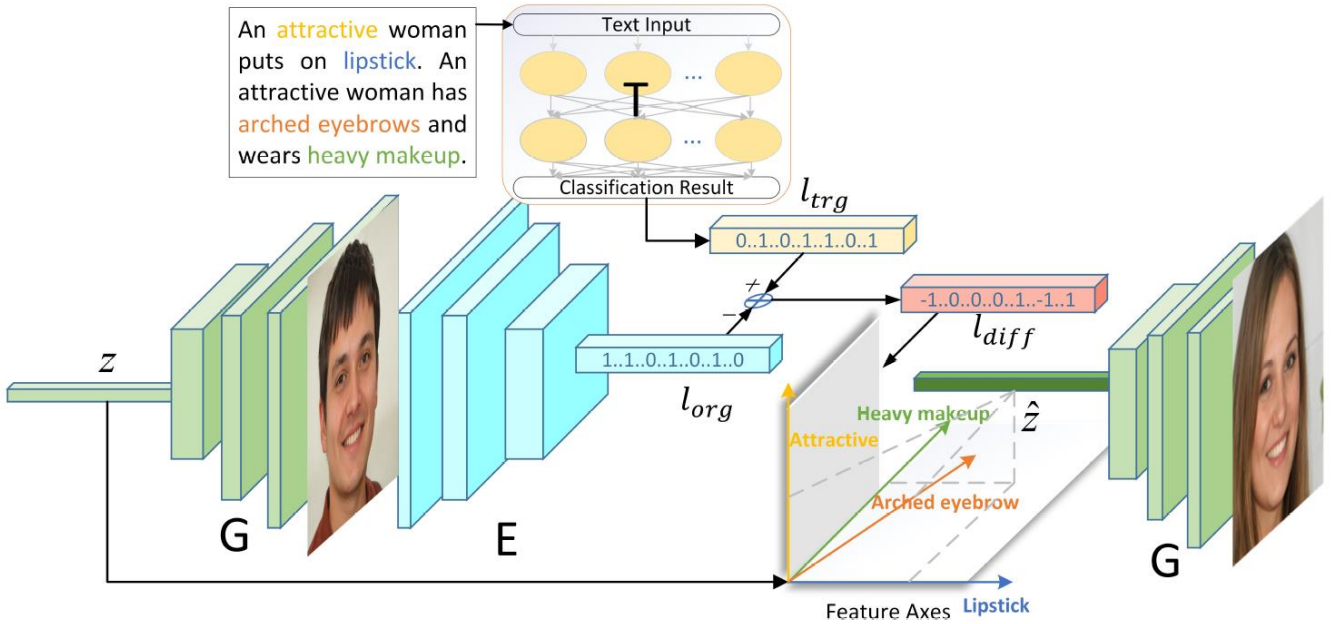
$$\mathbf{w}_k = \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}, (k = 1, 2, ...40) .$$

- 이 과정들을 거치고 나면, input noise vectors 의 방향을 갱신 할 수 있도록 도와주는 feature axes W 를 얻는다.

3.4. Noise Vector Manipulation

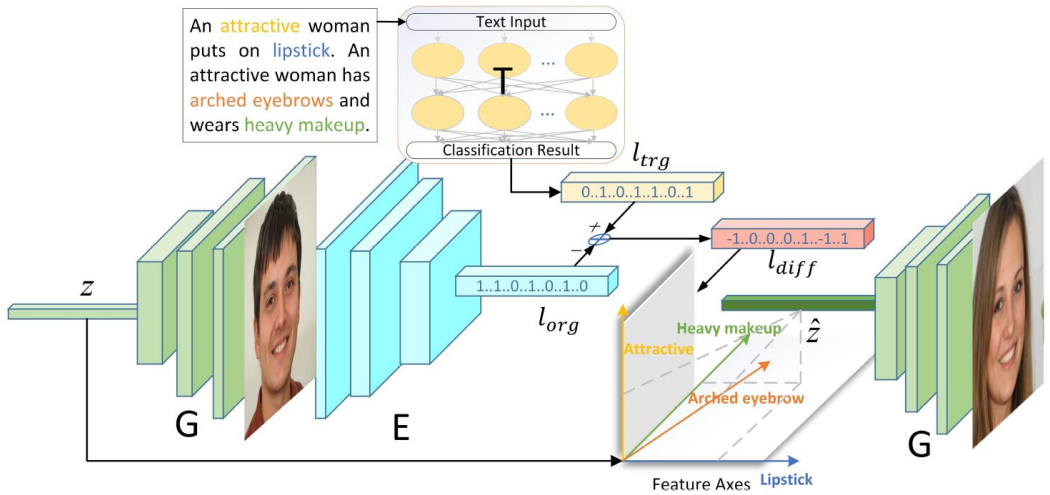
Noise Vector Manipulation

- Noise vectors 는 결과 이미지에 text corpus 로 설명된 특징들이 있는지 없는지 결정하기 때문에 noise vectors 를 조정하는 것은 중요하다.



- $\hat{z} = z + W \cdot l$
- l 은 feature 축에 따른 이동의 방향과 크기를 결정하는 column vector 이다.
- 네 개의 연산을 도입했다.

Noise Vector Manipulation



- Differentiation
  - Text classifier embedding output 은 l_{trg} 이고, 초기 random vector 로부터 예측된 embedding 은 $l_{org} = E(G(z))$ 이다.
  - 직관적으로, feature axes 를 따르는 noise vectors 의 이동을 안내하기 위해 l_{trg} 를 사용할 수 있다.
  - 그러나, l_{trg} 의 값의 범위는 $[0, 1]$ 으로, 이것은 모델이 반대 방향으로 features 를 만들 수 없다는 것을 의미한다.
  - 이것을 해결하기 위해, 우리는 differentiated embeddings l_{diff} 를 feature editing 을 안내하는 데에 사용한다.

$$l_{diff} = l_{trg} - l_{org}$$

  - Differentiated embeddings 의 값의 범위가 $[-1, 1]$ 이기 때문에 noise vectors 는 feature axes 의 양의 방향과 음의 방향 모두로 이동할 수 있다.

Noise Vector Manipulation

- Nonlinear Reweighting
  - Differentiated embeddings 에서, −1 또는 1의 값을 가진 labels 는 specified features 이다.
  - Specified features 를 강조할 필요가 있다. (이유는 아직 이해 x)
  - 강조를 위해 다음의 과정을 거친다.
    1. Differentiated embeddings 를 $[-1, 1]$ 에서 $[-\frac{\pi}{3}, \frac{\pi}{3}]$ 로 조정한다.
    2. Mapped differentiated embeddings 의 $\tan(\cdot)$ 를 계산한다.
    3. 그 결과, 범위의 끝에 도달한 값들은 더 높은 가중치를 얻을 것이다.
  - 본 논문의 경우 $\tan\left(\frac{\pi}{3}\right) = \sqrt{3}$ 이기 때문에, reweighted value 범위는 $[-\sqrt{3}, \sqrt{3}]$ 가 된다.

Noise Vector Manipulation

- Normalization
  - Noise vectors 가 정규 분포에서 샘플링 되었기 때문에, 확률 밀도가 높은 원점 근처에서 샘플링 될 확률이 높다.
  - 그러나, feature axes 를 따라 vectors 를 움직일 수록, vectors 와 원점 사이의 거리가 커지고, 그것은 생성된 이미지에서 더 많은 artifacts 를 만들 것이다.
  - 그래서 axes 를 따라 vectors 를 이동시킨 후에, vectors 를 renormalize 해야 한다.
  - 거리는 L_1 distance 를 사용한다.
  - 그러므로, noise vector $X = [x_1, x_2, \ldots, x_n]$ 를 $X' = [x'_1, x'_2, \ldots, x'_n]$ 로 renormalize 한다.

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{N=512} |\mathbf{x}_i|$$

$$\mathbf{x}'_i = \frac{\mathbf{x}_i}{\|\mathbf{x}\|_1} (i = 1, 2, ..., 512) \qquad (6)$$

Noise Vector Manipulation

- Feature lock
  - Face morphing process 를 더 안정적으로 만들기 위해, 특정 축을 따라 vectors 를 이동할 때마다 feature lock 단계를 거쳤다.
  - 즉, 모델은 vectors 가 이동된 축만 기본 축으로 사용한다.
  - 이런 방식으로, noise vectors 는 설명문에서 언급된 특징들로 lock 된다.

3.5. High Resolution Generator

High Resolution Generator

- Generator G 로 우리는 pre-trained model 인 StyleGAN2 를 사용한다.
- 이 generator 를 사용해서, 모델은 고화질 이미지를 합성할 뿐만 아니라 manipulated input vectors 로부터 원하는 features 를 만든다.

4. Experiments and Evaluation

Experiments and Evaluation

Figure 4: Images produced with single-sentence input. With fewer specified labels in the text, the model generates samples with higher variation.

Experiments and Evaluation

This old man has bags under his eyes. This chubby old man has a double chin. He has 5 o'clock shadow. He has receding hairline but bushy eyebrows. This fat man has straight gray hair. His face is pale with a pointy nose and big lips.

Figure 5: Image morphing via GAN of each group in the ablation study. (A) group with all operations applied (the default for TTF-HD); (B) group with reweighting, differentiation, and normalisation operations; (C) group with reweighting, differentiation operations; (D) group with the reweighting operation; and (E) group with no operations applied. We fix the noise vector input of each group. The figure shows the GAN morphing process from the randomly generated image on the left column to the final output on the right column.

Experiments and Evaluation

Table 1: Evaluation results of different models

Methods	***IS***	***CS****	***LPIPS***
TTF-HD (ours)	**1.117±0.127**	**0.664**	0.583±0.002
AttnGAN	1.062±0.051	0.511	——

*Maximum for each group.

Table 2: Ablation study evaluation results

Exp. Settings	**Evaluation Metrics**		
	IS	***CS****	***LPIPS***
Group A	1.122±0.043	0.754	**0.634±0.005**
Group B	1.116±0.080	0.739	0.608±0.005
Group C	**1.187±0.062**	**0.762**	0.603±0.005
Group D	1.101±0.095	0.683	0.521±0.006
Group E	1.102±0.033	0.706	0.532±0.005

*Maximum for each group

Thank you for listening