Each Attribute Matters: Contrastive Attention for Sentence-based Image Editing *(Supplementary Material)*

Liuqing Zhao *[1]
liuqingzhao@post.usts.edu.cn
Fan Lyu *[2]
fanlyu@tju.edu.cn
Fuyuan Hu [1]
fuyuanhu@mail.usts.edu.cn
Kaizhu Huang [3]
kaizhu.huang@xjtlu.edu.cn
Fenglei Xu [1]
xufl@mail.usts.edu.cn
Linyan Li †[4]
lilinyan@szjm.edu.cn

[1] Suzhou University of Science and Technology, Suzhou, China
[2] College of Intelligence and Computing, Tianjin University Tianjin, China
[3] Xi'an Jiaotong-Liverpool University, Suzhou, China
[4] Suzhou Institute of Trade and Commerce, Suzhou, China

*L.Zhao and F.Lyu share equal contribution.
† L.Li is the corresponding author.

This document provides supplementary material for the paper "Each Attribute Matters: Contrastive Attention for Sentence-based Image Editing" published on the British Machine Vision Conference (BMVC) 2021. In this material, we provide the further discussion and illustration of some details, and show more examples of SIE.

1 Evaluation Metrics

In our experiments, we use the **Fréchet Inception Distance (FID)** [7] and the **Learned Perceptual Image Patch Similarity (LPIPS)** [4] as the evaluation metrics.

The FID of an edited image compared to its origin is evaluated by passing it through a pre-trained Inception-v3 [1] and computing the distribution difference on the average pooled features. FID can be computed by

$$\mathrm{FID}_{(\mathbf{I},\hat{\mathbf{I}})} = \left\|\mu_{\mathbf{I}} - \mu_{\hat{\mathbf{I}}}\right\|_2^2 + \mathrm{Tr}\left(\Sigma_{\mathbf{I}} + \Sigma_{\hat{\mathbf{I}}} - 2\left(\Sigma_{\mathbf{I}}\Sigma_{\hat{\mathbf{I}}}\right)^{\frac{1}{2}}\right), \tag{1}$$

where $\mu_{\mathbf{I}}$ and $\mu_{\hat{\mathbf{I}}}$ represents the feature mean of the real image and the generated image. $\Sigma_{\mathbf{I}}$ and $\Sigma_{\hat{\mathbf{I}}}$ represents the covariance matrix of the features of the real image and the generated image. The smaller the FID value, the closer the distribution between generated image and real image.

We also use LPIPS to calculate the perceptual distance of two images. Traditionally, Perceptual distance [4] refers to the visual similarity of two images, the purpose of which is

to evaluate the similarity of two images by imitating the human visual senses. We extract feature from the l-th layer and unit-normalize it in the channel dimension, which we designate as $\mathbf{v}^l$ and $\hat{\mathbf{v}}^l \in \mathbb{R}^{c\times h\times w}$. h_l, w_l is the feature size in different layers. ω_l is equivalent to computing cosine distance. LPIPS can be computed by

$$\text{LPIPS}_{(\mathbf{I},\hat{\mathbf{I}})} = \sum_l \frac{1}{h_l w_l} \sum_{h,w} \|\omega_l \odot (\mathbf{v}^l - \hat{\mathbf{v}}^l)\|_2^2. \tag{2}$$

2 Sentence Parsing Strategy

In this paper, we propose a strategy to effectively parse sentence into different attributes, thus to facilitate the subsequent data augmentation and the construction of contrastive learning.

After the POS tagging of sentence $\mathcal{S}$, we put the corresponding lexical case of each sentence in $\mathcal{P}$. To classify the different attributes in a sentence, we have the following 5-step strategy. 1) Screening words for attributes; 2) Determine the adjective attribution of "bird has" case, divide “bird” into attributes and set the state of f_1, f_2 to 0; 3) If the word is a noun and not bird, the f_1 status is set to 1; 4) If the word is an adjective and is not followed by a conjunction, the f_2 status is set to 1; 5) When $f_1 \times f_2 = 1$, the attribute is divided. The detailed algorithm can be seen in Algorithm 1. Finally, we get the divided attributes $\hat{\mathcal{S}}$.

3 Discussion of hyperparameters

The generator and discriminator have trained alternatively by minimizing both the generator loss $\mathcal{L}_\text{G}$ and discriminator loss $\mathcal{L}_\text{D}$. In generator, $\mathcal{L}_\text{diff}$ control different attributes, $\mathcal{L}_\text{per}$ control the invariance of the background, $\mathcal{L}_\text{DAMSM}$ control text-image matching. In discriminator, $\mathcal{L}_\text{attr}$ discriminate the existence of attribute-level information.

$$\begin{aligned}\mathcal{L}_\text{G} = &-\frac{1}{2}E_{\hat{\mathbf{I}}\sim P_\text{G}}[\log(\text{D}(\hat{\mathbf{I}}))] - \frac{1}{2}\mathbb{E}_{\hat{\mathbf{I}}\sim P_G}[\log(\text{D}(\hat{\mathbf{I}},\mathcal{S}))] \\ &+ \lambda_1\mathcal{L}_\text{diff} + \lambda_2\mathcal{L}_\text{per} + \lambda_3\mathcal{L}_\text{DAMSM}\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{L}_\text{D} = &-\frac{1}{2}\mathbb{E}_{\mathbf{I}\sim \text{P}_\text{data}}[\log(\text{D}(\mathbf{I}))] - \frac{1}{2}\mathbb{E}_{\hat{\mathbf{I}}\sim P_G}[\log(1-\text{D}(\hat{\mathbf{I}}))] \\ &-\frac{1}{2}\mathbb{E}_{\hat{\mathbf{I}}\sim \text{P}_\text{G}}[\log(1-D(\hat{\mathbf{I}},\mathcal{S}))] - \frac{1}{2}\mathbb{E}_{\hat{\mathbf{I}}\sim \text{P}_\text{G}}[\log(\text{D}(\mathbf{I},\mathcal{S}))] \\ &+ \lambda_4\mathcal{L}_\text{attr}\end{aligned} \tag{4}$$

The proposed algorithm is governed by four hyperparameters: λ_1, λ_2 and λ_3 are used in the generator to balance the generation of different attributes and to preserve irrelevant backgrounds. Our model is based on the AttnGAN [3] model, so for the hyperparameter λ_3, we follow its initial value and do not adjust it. In the discriminator, λ_4 to control whether each attribute is present in the image or not. Table 1 shows the sensitivity analysis for hyperparameters using the CUB dataset. As a rule of thumb, we try from 1 and calculate the FID and LPIPS values for each model. We found that the models work better when in the range of 0.5 to 1.

λ_1	λ_2	λ_3	λ_4	FID ↓	LPIPS ↓
1	1	1	1	23.97	0.7085
0.5	0.5	1	0.5	22.75	0.7073
0.5	0.5	1	1	21.26	0.7046
0.5	1	1	1	22.09	0.7067
0.7	**0.6**	**1**	**0.9**	**20.08**	**0.6893**
1.5	1.5	1	1.5	24.08	0.7091

Table 1: Hyperparameter analysis.

4 Additional SIE Examples

In Fig. 1 2 3, we show a qualitative comparison of the models on the COCO, CUB dataset.

References

[1] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[2] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

[3] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018.

[4] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

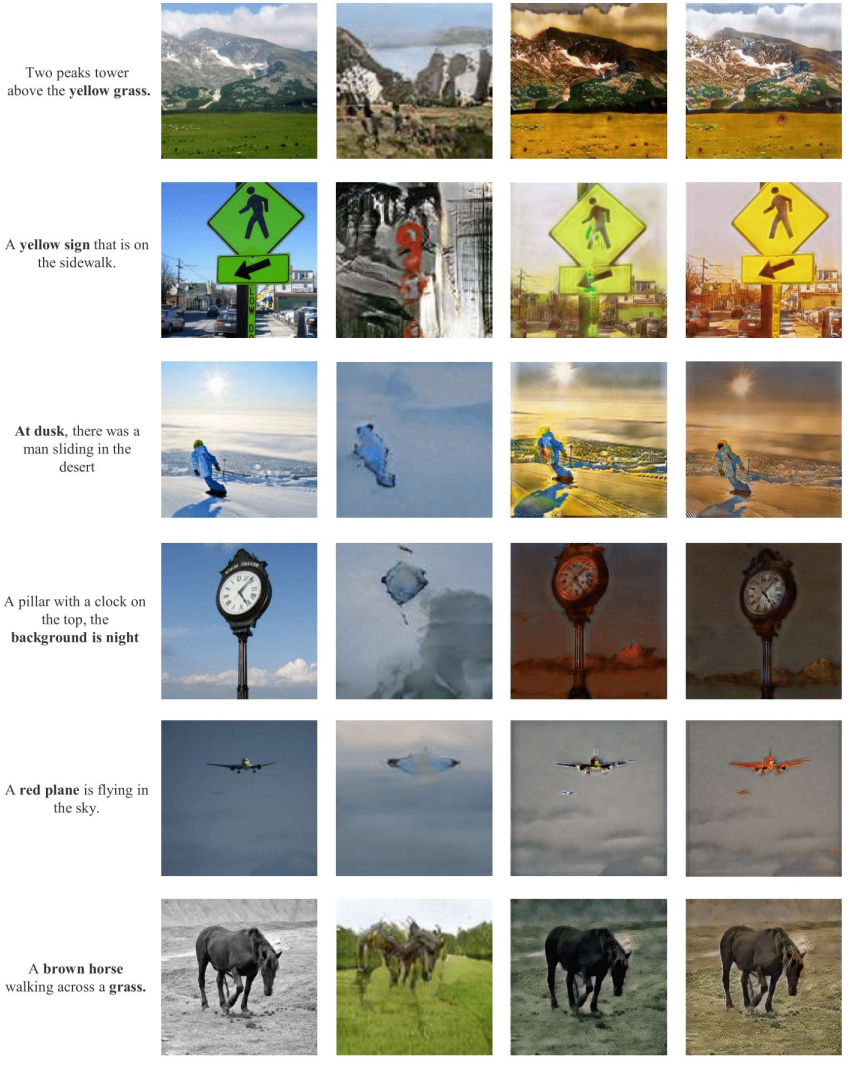

Figure 1: Additional comparison results between TAGAN, ManiGAN, and Ours on the COCO dataset.

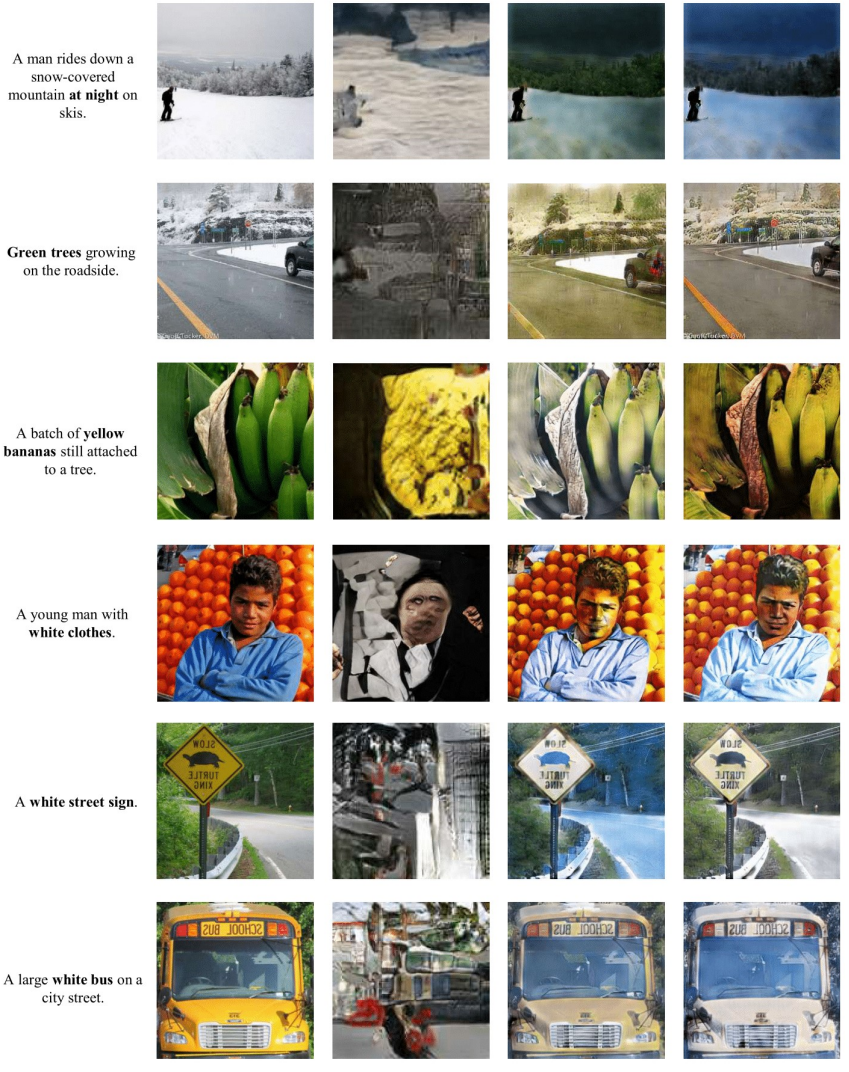

Figure 2: Additional comparison results between TAGAN, ManiGAN, and Ours on the COCO dataset.

Figure 3: Additional comparison results between SISGAN, TAGAN, DMIT, ManiGAN, and Ours on the CUB dataset.

Algorithm 1: Sentence Parsing method

Input: Sentence $\mathcal{S} = \{w_1, \cdots, w_N\}$, Status control symbols f_1 =0, $f_2 = 0$, Counters $n = 0, m = 0$.

Output: Parsed Sentence $\hat{\mathcal{S}} = \{\mathcal{A}_1, \cdots, \mathcal{A}_M\}$, $\mathcal{A} = \{w_i\}$.

$\mathcal{P} = \{a_1, \cdots, a_N\} \leftarrow$ POS tagging $(\mathcal{S})$;
for $i \leftarrow 0$ **to** $len(\mathcal{S})$ **do**
  // 1.Screening words for attributes
  if $\mathcal{P}_i \in$ {NN, NNS, JJ} *or* $w_i \in$ {"has", "with", "and"} **then**
    // 2.Judging the adjective attribution in the subject case
    if w_i = "bird" *and* $w_{i+1} \in$ {"has", "with"} **then**
      $\mathcal{A}_n \leftarrow \{w_i, \cdots\}$;
      $f_1 \leftarrow 0, f_2 \leftarrow 0$;
      $n = n + 1$;
    end
    // 3.Judging the noun case
    if $\mathcal{P}_i$ = "NN" *or* $\mathcal{P}_i$ = "NNS" *and* $w_i \neq$ "bird" **then**
      $\mathcal{A}_n \leftarrow w_i$;
      $f_1 \leftarrow 1$;
    end
    // 4.Judging the adjective attribution in conjunctive cases
    if $\mathcal{P}_i$ = "JJ" and $w_{i+1} \neq$ "and" **then**
      $\mathcal{A}_n \leftarrow w_i$;
      $f_2 \leftarrow 1$;
    end
    // 5.Classifying attributes
    if $f_1 \times f_2 = 1$ **then**
      $\mathcal{A}_n \leftarrow \{w_i, \cdots\}$;
      $f_1 \leftarrow 0, f_2 \leftarrow 0$;
      $n = n + 1$
    end
  end
  $\hat{\mathcal{S}}_m \leftarrow \mathcal{A}_n$
  $m = m + 1$
end