

Target-Free Text-Guided Image Manipulation

Wan-Cyuan Fan¹, Cheng-Fu Yang², Chiao-An Yang³, Yu-Chiang Frank Wang^{1, 4}

¹National Taiwan University

²UCLA

³Purdue University

⁴NVIDIA

r09942092@ntu.edu.tw

Abstract

We tackle the problem of target-free text-guided image manipulation, which requires one to modify the input reference image based on the given text instruction, while no ground truth target image is observed during training. To address this challenging task, we propose a Cyclic-Manipulation GAN (cManiGAN) in this paper, which is able to realize *where* and *how* to edit the image regions of interest. Specifically, the image editor in cManiGAN learns to identify and complete the input image, while cross-modal interpreter and reasoner are deployed to verify the semantic correctness of the output image based on the input instruction. While the former utilizes factual/counterfactual description learning for authenticating the image semantics, the latter predicts the “undo” instruction and provides pixel-level supervision for the training of cManiGAN. With such operational cycle-consistency, our cManiGAN can be trained in the above weakly supervised setting. We conduct extensive experiments on the datasets of CLEVR and COCO, and the effectiveness and generalizability of our proposed method can be successfully verified. Project page: sites.google.com/view/wancyuanfan/projects/cmanigan.

Introduction

Image manipulation by text instruction (or text-guided image manipulation) aims to edit the input reference image based on the given *instruction* that describes the desirable modification to the image. This task not only benefits a variety of applications including computer-aided design (El-Nouby et al. 2019; Viazovetskyi, Ivashkin, and Kashin 2020), face generation (Xia et al. 2021b,a) and image editing (Zhang et al. 2021; Shetty, Fritz, and Schiele 2018; Li et al. 2020a; Wang et al. 2021; Patashnik et al. 2021; Li et al. 2020b), the developed algorithm can be further applied as a data augmentation technique for learning deep neural networks. In addition to the need to output high-quality images, the main challenges in text-guided image manipulation are to identify “*where*” and to know “*how*” to edit the image based on the given instruction. In other words, how to bridge the gap between semantic and linguistic information during image manipulation process requires the efforts from researchers in related fields.

Existing text-to-image manipulation works can be divided into two categories: *object-centric image editing* and *scene-level image manipulation* (Shetty, Fritz, and Schiele 2018). For object-centric image editing (Choi et al. 2018; Li et al. 2019, 2020b,a), existing works focus on modifying visual attributes (e.g., color or texture) of particular objects (e.g., face or bird) in the image, or to change its style (e.g., age or expression) to match the given *description* (not necessarily *instruction*). With such image-description pairs observed during training, the text-guided editing process can be simply achieved by mapping the information between the images and the corresponding descriptions. While ground truth target image is not necessarily required, such models require descriptions for both images before and after editing.

As for scene-level image manipulation (El-Nouby et al. 2019; Shetty, Fritz, and Schiele 2018; Zhang et al. 2021; Dhama et al. 2020), its goal is to reorganize the composition of the input image (e.g., moving, adding, and removing objects in the images). Since the input image might contain multiple objects in the scene, to localize “*where*” to edit would be a difficult task to handle. Moreover, instead of changing attributes of a given object, the model needs to generate objects or introduce a background on the location of interest. Thus, Zhang et al. (Zhang et al. 2021) decompose the above manipulation process into two stages: localization and generation. With target images as supervision, TIMGAN (Zhang et al. 2021) is trained to manipulate the reference image with visual realism and semantic correctness. However, since the ground truth target image might *not* always be available, Adversarial Scene Editing (ASE) (Shetty, Fritz, and Schiele 2018) chooses a weakly supervised setting with image-level labels as weak guidance. Nevertheless, ASE only allows one to *remove* an object in the scene and cannot easily be applied to operations like adding or changing an attribute.

In this paper, we propose a cyclic-Manipulation GAN (cManiGAN) for target-free image manipulation. Due to the absence of ground truth target image during training, it is extremely challenging to identify *where* and *how* to edit the input reference image, so that the output would be semantically correct. To tackle the above two obstacles, our cManiGAN exploits self-supervised learning for enforcing the semantic correctness, while pixel-level guidance can be automatically observed. More specifically, the image editor of

Methods	Input data				Manipulation type		
	Instruction	Description	GT image	Auxiliary info	Change attribute	Remove	Add
ManiGAN (Li et al. 2020a)	-	✓	No	-	✓	-	-
TediGAN (Xia et al. 2021a)	-	✓	No	-	✓	-	-
ASE (Shetty, Fritz, and Schiele 2018)	-	-	No	Image-level labels	-	✓	-
GeNeVa (El-Nouby et al. 2019)	✓	-	Yes	-	-	-	✓
TIM-GAN (Zhang et al. 2021)	✓	-	Yes	-	✓	✓	✓
Ours	✓	-	No	Image-level labels	✓	✓	✓

Table 1: Comparisons of recent approaches on text-guided image manipulation. Note that GT image indicates the need of ground-truth target image during training.

cManiGAN learns to locate/edit the image region of interest with global and local semantics observed. And, the modules of cross-modality interpreter and reasoner are deployed in cManiGAN. The former is introduced to verify the image semantics via factual/counter-factual description learning, while the latter infers the “undo” instruction and offers pixel-level supervision. As detailed later, the above design uniquely utilizes cross-modal cycle consistency and allows the weakly-supervised training our of cManiGAN.

Related Works

Object-centric Image Manipulation by Text Instruction.

ControlGAN (Li et al. 2019) is an end-to-end trainable network for synthesizing high-quality images, with image regions fitting the given descriptions. Li et al. propose (Li et al. 2020a,b), containing an affine combination module (ACM) and a detail correction module (DCM), which manipulate image regions based on both input text and the desired attributes (e.g., color and texture). Liu et al. (Liu et al. 2020) utilize unified visual-semantic embedding space so that manipulation can be achieved by performing text-guided vector arithmetic operations. Recently, Xia et al. (Xia et al. 2021a) apply StyleGAN to edit the reference image via instance-level optimization. With the given text as guidance, its produced image would be close to the reference input in the embedding space.

Scene-level Image Manipulation by Text Instruction.

El-Nouby et al. (El-Nouby et al. 2019) propose GeNeVa for the serial story image manipulation, which sequentially predicts new objects based on the associated descriptions to a story scene background. Dharmo et al. propose SIMSG (Dharmo et al. 2020), which encodes image semantic information into a given scene graph for manipulation purposes. Recently, TIM-GAN (Zhang et al. 2021) decomposes the manipulation process into localization and generation. The introduced Routing-Neurons network in the generation exhibits the ability to dynamically adapt different learning blocks for different complex instructions, better capturing text information and thus with improved manipulation ability. As noted earlier, existing methods for scene-level image manipulation require reference-target training image pairs (i.e., a fully supervised setting). While methods such as ManiGAN (Li et al. 2020a) and TediGAN (Xia et al. 2021a) do not observe the target images during training, these methods are generally applied to object-centric images and cannot be easily generalized to perform scene-level manipulation.

To alleviate the above concern, Shetty et al. (Shetty, Fritz, and Schiele 2018) introduce ASE, allowing users to remove an object in a scene-level image while not requiring pairwise training data. Instead of producing an entire image output based on given text (Li et al. 2019; Reed et al. 2016) and scene graphs (Herzig et al. 2020; Johnson, Gupta, and Fei-Fei 2018; Yang et al. 2021), ASE focuses on generating the background of a specific area on the image to smoothly remove the target object. As listed in Table 1, we compare the properties of recent image manipulation approaches and highlight the practicality of ours.

Methodology

Notations and Algorithmic Overview

In this work, one is given a reference image I_r and the associated instruction \mathcal{T} , describing where and how to edit I_r . Without observing the ground truth target image I_t , our goal is to produce an image I_g matching \mathcal{T} (i.e., with the desirable layout and/or attributes).

To tackle this problem, we propose a GAN-based framework of Cyclic Manipulation GAN (cManiGAN), as shown in Fig. 1(a). Given I_r and \mathcal{T} , our cManiGAN has an image editor (generator) G , which consists of a spatial-aware localizer L and an image in-painter P . The former is deployed to identify the target object/attributes of interest by producing a binary mask M , and the latter is to complete the output image I_g accordingly. To enforce the correctness of the visual attribute and the location of the generated object, a cross-modal *interpreter* \mathcal{I} is introduced in cManiGAN, which learns to distinguish between factual and synthesized counterfactual descriptions. Moreover, an instruction *reasoner* R is deployed to infer the “undo” instruction \mathcal{T}' from \mathcal{T} . With this “undo” instruction, a cyclic-consistent training scheme can be conducted which provides pixel-level supervision from the recovered image output.

Following (Shetty, Fritz, and Schiele 2018), we utilize the image-level labels (i.e., O_r and O_t) from the reference and target images as weak supervision. Note that such labels are practically easy to obtain, since they can be produced by pre-trained classifiers or by rule-based language models like (Manning et al. 2014) to infer the labels from I_r and \mathcal{T} (as discussed in the supplementary materials).

Image Editor G

An overview of the proposed Cyclic-Manipulation GAN (cManiGAN) is illustrated in Fig. 1(a). The generator, or

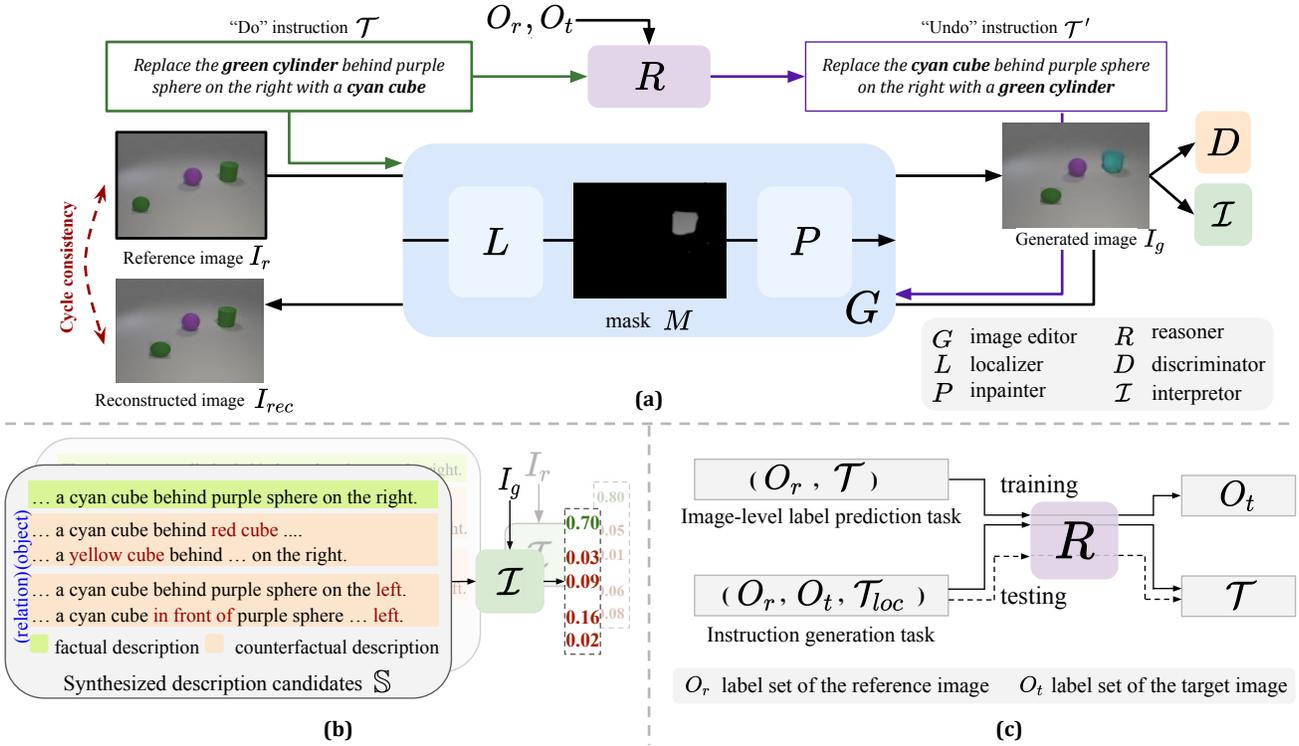


Figure 1: (a) Architecture of cManiGAN, which consists of generator G (with localizer L and inpainter P), discriminator D , cross-modal interpreter \mathcal{I} , and reasoner R . Note that O_r and O_t are image-level labels for the reference and target images, respectively. (b) Cross-modal interpreter \mathcal{I} in (a), which authenticates the output image via factual/counterfactual descriptions. (c) Reasoner R deployed in (a) to produce the undo instruction for cross-modal cycle consistency. Note that \mathcal{T}_{loc} is the adverbs of place part of instruction \mathcal{T} . Please see Methodology for detailed discussions for each module.

image editor G , aims to modify the given reference image I_r based on the instruction \mathcal{T} and produces the manipulated result I_g . We now detail the design and learning objectives for the generator G .

Localizer L . As shown in Fig. 1(a), a spatial-aware localizer L is deployed in the first stage of G to identify the target object/location in I_r . With the adverb \mathcal{T}_{loc} related to locations extracted from \mathcal{T} via CoreNLP (Manning et al. 2014), it can be further encoded as $f_{loc}^{\mathcal{T}}$, describing the embedding of the location of interest with a pre-trained BERT (Devlin et al. 2018). Together with I_r , L learns to mask out the object of interest by producing a binary mask $M = L(I_r, \mathcal{T}_{loc})$. More precisely, this is achieved by having L perform cross-modal attention between $f_{loc}^{\mathcal{T}}$ and the feature map of I_r , followed by a mask decoder to produce M .

Unfortunately, it would be difficult to verify the correctness of the aforementioned mask without the presence of the target image. During the training of our cManiGAN, we apply a standard classification objective $\mathcal{L}_{in}^L = \mathcal{L}_{CE}(\text{MLP}(E(M \odot I_r)), y_{in}^r)$ for the masked part, and the multi-label classification loss $\mathcal{L}_{out}^L = \mathcal{L}_{BCE}(\text{MLP}(E((1 - M) \odot I_r)), y_{out})$ for the unmasked region. We have E as the feature extractor (e.g., VGG (Simonyan and Zisserman

2014)), y_{in}^r and y_{out} denote the one/multi-hot label vectors indicating object category/categories in the masked/unmasked parts in I_r and I_t , respectively. In the above derivation, MLP denotes multi-layer perceptron, and \odot indicates element-wise dot product. And, \mathcal{L}_{CE} and \mathcal{L}_{BCE} represent the cross-entropy/binary-cross-entropy losses. Thus, the objective for learning the localizer L is calculated by summing up \mathcal{L}_{in}^L and \mathcal{L}_{out}^L . Thus, with the design and deployment of the Localizer, we are able to enforce the generator to manipulate the location of interest only. As later verified in our ablation studies, this allows our model to improve the generating quality and the feasibility of image manipulation under a weakly supervised setting.

Image Inpainter P . As the second stage in G , we have an image inpainter P which takes the text feature $f_{how}^{\mathcal{T}}$ extracted from \mathcal{T} by pre-trained BERT and the masked input $(1 - M) \odot I_r$ for producing I_g . In addition to the standard GAN loss (Goodfellow et al. 2014) for the generator G with a discriminator D deployed, we also calculate the reconstruction loss (i.e., mean squared error) $\mathcal{L}_{rec}^P = \mathcal{L}_{MSE}((1 - M) \odot I_r, (1 - M) \odot I_g)$, preserving the content of unmasked image regions $(1 - M) \odot I_g$. Moreover, with an auxiliary classifier \mathcal{C} jointly trained with the discriminator, we calculate the following classification losses:

$\mathcal{L}_{out}^P = \mathcal{L}_{BCE}(\mathcal{C}((1 - M) \odot I_g), y_{out})$, ensuring the semantic correctness of the unmasked image regions, and $\mathcal{L}_{in}^P = \mathcal{L}_{CE}(\mathcal{C}((M \odot I_g), y_{in}^g))$ to enforce that of the *manipulated* object in $M \cdot I_g$. Note that y_{in}^g denotes the one-hot label vector indicating the ground truth category of the object within the target location in I_g .

With the above design, we have the total objective for P as the sum over \mathcal{L}_{GAN} , \mathcal{L}_{rec}^P , \mathcal{L}_{in}^P , and \mathcal{L}_{out}^P , with both image global and local authenticities enforced by D (Iizuka, Simo-Serra, and Ishikawa 2017). Please refer to the supplementary materials for the full objectives and implementation details.

Cross-Modal Interpreter \mathcal{I}

For training target-free image manipulation models, how to preserve both visual and semantic correctness (e.g., visual attributes and spatial relationship) of the output image would be a challenging task. Thus, in addition to the above G and D modules, we introduce a cross-modal interpreter \mathcal{I} in our cManiGAN for achieving this goal with additional word-level correctness enforced.

Learning from Factual/counterfactual Descriptions. As shown in Fig. 1(b), given an input image, our cross-modal interpreter \mathcal{I} learns to discriminate the factual description among multiple synthesized description candidates. This would enforce the manipulated image to be with sufficient visual and semantic correctness. To generate factual and counterfactual descriptions of an image, we utilize \mathcal{T} and the labels for reference and target images (i.e., $O_r = \{o_1, o_2, \dots, o_n\}$ and $O_t = \{o_1, o_2, \dots, o_m\}$). We note that, with n and m represent the associated total numbers of objects, we have $|m - n| \leq 1$ since each instruction is considered to manipulate a single object. Thus, we define a basic description template as follows: ‘‘There is a [OBJ] [LOC]’’, where [OBJ] can be replaced by the object label, and [LOC] indicates the adverbs of place to describe where [OBJ] is.

With the above definitions, we synthesize the factual description \mathcal{S}^f by replacing the [LOC] with adverbs of the place of the given instruction, extracted by CoreNLP (Manning et al. 2014). We then replace the [OBJ] with the objects o_r^f and o_t^f for the reference and target images to generate the corresponding descriptions separately. The o_r^f and o_t^f can be identified by comparing the difference between O_r and O_t with the following three principles:

- if $n > m$, $o_r^f = O_r - O_t$ and o_t^f is NONE.
- if $n = m$, $o_r^f = O_r - (O_r \cap O_t)$ and $o_t^f = O_t - (O_r \cap O_t)$.
- if $n < m$, o_r^f is NONE, and $o_t^f = O_t - O_r$.

Note that NONE denotes the dummy category, which implies no object of interest at that location.

As for synthesizing the counterfactual descriptions, we collect object tokens (e.g., green sphere and red cube in the CLEVR dataset) and relation tokens (e.g., in front of and behind) by applying NLTK tools (Bird, Klein, and Loper 2009) on the factual ones. With these tokens, each object/relation counterfactual descriptions \mathcal{S}_i^c (i denotes the counterfactual description index) can be generated by randomly replacing the existing object/relation tokens

from the factual descriptions with other non-existing tokens. As a result, a set of counterfactual descriptions $\mathbb{S}^c = \{\mathcal{S}_1^c, \mathcal{S}_2^c, \dots, \mathcal{S}_N^c\}$ can be obtained by repeating the above process. Note that N is the total number of counterfactual descriptions.

Authenticating Semantic Correctness of I_g . With the interpreter \mathcal{I} taking the generated image I_g and a set of descriptions $\mathbb{S} = \{\mathcal{S}^f, \mathcal{S}_1^c, \mathcal{S}_2^c, \dots, \mathcal{S}_N^c\}$ (i.e., one factual description and N counterfactual ones) as the inputs, our cManiGAN is able to assess the semantic correctness of I_g by calculating the cross-modal matching scores \hat{y} between the generated image I_g and each of the descriptions in \mathbb{S} . For the architecture of \mathcal{I} , it can be viewed as a cross-modal alignment module Γ (e.g., ViLBERT (Lu et al. 2019) or a word-level discriminator in LWGAN (Li et al. 2020b)), which takes an image and language description/caption as inputs for producing the associated matching scores. Thus, the output scores \hat{y} can be calculated as:

$$\hat{y} = \mathcal{I}(I, \mathbb{S}) = [\Gamma(I, \mathcal{S}^f), \Gamma(I, \mathcal{S}_1^c), \Gamma(I, \mathcal{S}_2^c), \dots, \Gamma(I, \mathcal{S}_N^c)]. \quad (1)$$

To train our two-stage editor with the interpreter \mathcal{I} , we take \mathcal{I} as an auxiliary classifier along with D . Similar to ACGAN (Odena, Olah, and Shlens 2017), the loss function $\mathcal{L}_{\mathcal{I}}$ of the interpreter can thus be formulated as follows:

$$\mathcal{L}_{\mathcal{I}} = \mathcal{L}_{CE}(\mathcal{I}(I_r, \mathbb{S}_r), y) + \mathcal{L}_{CE}(\mathcal{I}(I_g, \mathbb{S}_t), y). \quad (2)$$

Note that \mathbb{S}_r and \mathbb{S}_t denote the synthesized description sets for reference and target images, respectively. Also, y is one-hot vector with the only nonzero entry associated with the factual description.

Reasoner R

Operational Cycle Consistency. With the deployment of the generator G , discriminator D , and the interpreter \mathcal{I} , our cManiGAN is able to produce images with semantic authenticity preserved at the *image* level. To further enforce the correctness at the *pixel* level, we further introduce a reasoner R in cManiGAN. As depicted in Fig. 1(a), this reasoner R is designed to predict the undo instruction \mathcal{T}' from I_r and \mathcal{T} (together with O_r and O_t), which outputs the reconstructed version I_{rec} and observe operational cycle consistency by minimizing the difference between I_r and I_{rec} . Thus, this consistency objective \mathcal{L}_{cyc} is calculated as:

$$\mathcal{L}_{cyc} = \mathcal{L}_{MSE}(I_r, I_{rec}) + \mathcal{L}_{perc.}(I_r, I_{rec}), \quad (3)$$

where $I_{rec} = G(G(I_r, \mathcal{T}), \mathcal{T}')$. Also, \mathcal{L}_{MSE} and $\mathcal{L}_{perc.}$ represent the mean squared error and the perceptual loss (Johnson, Alahi, and Fei-Fei 2016), respectively.

Learning from Sequence-to-sequence Models. Since the undo instruction is a textual sequence, we approach this reasoning task by solving a sequence-to-sequence learning problem and adopt the recent model of T5 (Raffel et al. 2019) as the base model. However, since sequence models like T5 are trained on the language crawled corpus (Raffel et al. 2019), they cannot be directly applied for text-guided

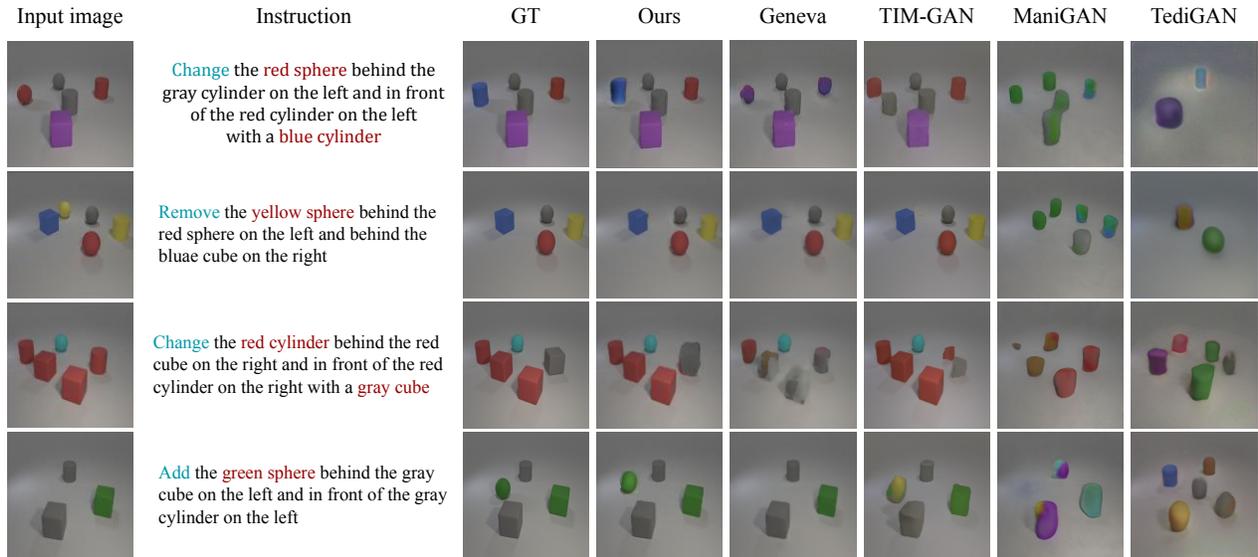


Figure 2: Qualitative evaluation on the CLEVR dataset. Each row shows the input reference image, instruction, ground truth (target) image and those generated by different methods. Note that GeNeVa (El-Nouby et al. 2019) and TIM-GAN (Zhang et al. 2021) require target images during training. And, ManiGAN (Li et al. 2020a) and TediGAN (Xia et al. 2021a) are mainly designed to tackle object-centric image data.

image manipulation. Thus, as shown in Fig. 1(c), we design two learning tasks which adapt T5 for our reasoning task. First, we consider the *image-level labels prediction* task, which predicts the labels O_t by observing O_r and \mathcal{T} , aiming at relating the changes in image-level labels. To equip R with the ability to express its observation in terms of desirable instructions, we consider *instruction generation* as the second fine-tuning task, with the goal to synthesize the full instruction by inputting O_r , O_t and the adverbs of place part of instruction \mathcal{T}_{loc} (extracted by CoreNLP (Manning et al. 2014) as noted earlier). With these two fine-tuning tasks, our reasoner R is capable of inferring the undo instruction \mathcal{T}' by observing O_t , O_r and \mathcal{T} .

We now detail the learning process for our reasoner R . As shown in Fig. 1(c), to realize the sequence-to-sequence training scheme, we first convert image labels into pure text format with the subject-verb-object (SVO) sentence structure. Take $O_r = \{\text{purple sphere, green cube}\}$ for example, we have the text format as “The labels for reference image contains purple sphere, green cube.” Also, we denote the \mathcal{T}_r^O and \mathcal{T}_t^O as the text format of O_r and O_t , respectively. With the labels described in text format, we construct the input text sequences $\hat{\mathcal{T}}$ for the fine-tuning tasks. For *image-level prediction*, we have $\hat{\mathcal{T}} = \mathcal{T}_r^O \oplus \mathcal{T}$ serve as the input context, the T5 model is learned to predict \mathcal{T}_t^O . As for the second task of *instruction generation*, the input context would be $\hat{\mathcal{T}} = \mathcal{T}_r^O \oplus \mathcal{T}_t^O \oplus \mathcal{T}_{loc}$, and the output would be the given instruction \mathcal{T} , where \oplus is the concatenation on text. Note that the input text sequence for each fine-tuning task can be further created by combining a task-specific (text) prefix (e.g., what is the instruction) with the context according to the task of interest. Please refer to the supplementary materials for

details and more training examples.

With the above designs, we impose the conventional sequence-to-sequence objective \mathcal{L}_{s2s} (Raffel et al. 2019) to fine-tune the T5 model. Thus, the objective for learning R can be formulated as follows:

$$\mathcal{L}_R = \mathcal{L}_{s2s}(R(\mathcal{T}_r^O \oplus \mathcal{T}_t^O \oplus \mathcal{T}_{loc}), \mathcal{T}) + \mathcal{L}_{s2s}(R(\mathcal{T}_r^O \oplus \mathcal{T}), \mathcal{T}_t^O). \quad (4)$$

Once the above model is fine-tuned as our reasoner R , the undo instruction \mathcal{T}' can be directly inferred by observing input test sequence as $\hat{\mathcal{T}} = \mathcal{T}_{loc} \oplus \mathcal{T}_r^O \oplus \mathcal{T}_t^O$ with the labels for the reference and target images swapped.

As illustrated in Fig. 1(a), operational cycle consistency can be observed during the training of cManiGAN, providing additional desirable pixel-level guidance. For complete learning details (including pseudo code) of our cManiGAN, please refer to the supplementary material.

Experiment

Datasets

CLEVR The CLEVR dataset (Johnson et al. 2017) is created for multimodal learning tasks such as visual question answering, cross-modal retrieval, and iterative story generation. We consider the synthesized version of CLEVR as TIM-GAN (Zhang et al. 2021) did, which contains a total of 24 object categories (red cube, blue sphere, and cyan cylinder, etc.) with about 28.1K/4.6K paired synthesized images for training/validation. Each training sample includes two paired images (reference image and target image (for evaluation only)) and an instruction describing where and how to manipulate the reference image.

Operation	Type 1: remove + add					Type 2: attribute change / shape				
	FID ↓ IS ↑	image acc (%)	In-mask acc (%)	Interp. acc (%)	R@1 R@5	FID ↓ IS ↑	image acc (%)	In-mask acc (%)	Interp. acc (%)	R@1 R@5
Upper bound	- -	99.25	88.66	67.16	72.12 99.77	- -	98.71	90.91	67.19	96.27 99.85
GeNeVa [†]	54.80 2.336	92.93	40.08	34.27	33.32 79.23	52.91 2.017	88.65	7.18	11.18	64.17 76.75
TIM-GAN [†]	43.38 2.192	93.40	25.50	38.17	33.72 80.81	54.66 2.122	90.05	4.67	10.79	58.73 76.37
ManiGAN	168.5 2.390	75.68	20.12	0.88	0.01 0.09	170.12 2.234	73.78	2.3	0.42	0.08 0.17
TediGAN	172.2 2.760	69.60	26.07	4.02	0.01 0.49	168.1 2.672	69.47	2.46	0.76	0.04 0.64
Ours	45.88 2.214	93.59	43.01	40.85	47.95 94.04	38.26 2.210	93.18	39.18	33.74	87.46 94.01

Table 2: Quantitative evaluation on CLEVR. Note that R@N indicates the recall of the true target image in the top-N retrieved images. [†] denotes methods requiring target images for training. The numbers in bold indicate the best scores, and those with underlines denote the second highest ones.

	FID ↓ IS ↑	image acc (%)	Inside-mask acc (%)	Interp. acc (%)
Upper bound	- -	91.47	92.49	68.71
Ours	166.18 4.64	86.04	17.17	13.54
ASE [†]	132.04 6.37	86.99	41.66	33.34
Ours [†]	104.77 7.21	89.73	50.03	46.20

Table 3: Quantitative results on COCO. [†] denotes only the “remove” operation is considered during evaluation.

COCO The COCO dataset (Lin et al. 2014) contains 118k real-world scene images for training with a total of 80 thing categories (car, dog, etc.). For simplicity, we consider a sampled COCO dataset containing 20 object categories (overlapped with Pascal-VOC (Everingham et al. 2015)) with about 12K/3K samples for training/validation. Since the target images are not available for COCO, only a reference image and an instruction are included in a training/validation sample (see supplementary material for the details).

Note that three types of manipulations/operations, i.e., “add”, “remove”, and “change”, are considered in both datasets. We will make the datasets publicly available for reproduction and comparison purposes.

Qualitative Evaluation

We compare our proposed cManiGAN with recent models, including GeNeVa (El-Nouby et al. 2019), TIM-GAN (Zhang et al. 2021), ManiGAN (Li et al. 2020a) and TediGAN (Xia et al. 2021a), with example results shown in Fig. 2. From this figure, we observe that the outputs of the attribute-based methods, such as ManiGAN and TediGAN, were not able to locate proper image regions for manipulation and even failed to preserve the image structure or details. As for the structure-based methods (i.e., GeNeVa and TIM-GAN), even though the structure of the reference image was preserved, these methods fail to comprehend the complex input instruction and lack the ability to manipulate the image with visual and semantic correctness. Take the third case in Fig. 2 for example, GeNeVa incorrectly changed the visual attributes of two non-target objects. On

the other hand, our cManiGAN was able to generate consistent outputs following the given instructions. For more qualitative results, please refer to our supplementary materials.

Quantitative Evaluation

To quantify and compare the performances between different models, two metrics are applied: Fréchet inception distance (FID) and inception score (IS). Moreover, the following four different metrics are utilized for evaluation:

- (1) **Image classification accuracy (Image acc)** measures whether the objects in the generated image match the labels of the target image.
- (2) **Inside mask classification accuracy. (In-mask acc)** evaluates whether the generated object in the masked part can be recognized by a pre-trained classification model.
- (3) **Interpreter accuracy (Interp. acc).** measures whether the generated image semantically matches its factual description via a cross-modal interpreter, which is pre-trained on the reference image I_r and its corresponding description set \mathbb{S}_r .
- (4) **Retrieval score (RS)** evaluates the manipulation correctness of the manipulation by applying the existing text-guided image retrieval method of TIRG (Vo et al. 2019). Following TIM-GAN (Zhang et al. 2021), we report RS@N, where N indicates the recall of the ground-truth image in the top-N retrieved images.

Quantitative Comparisons Image Quality and Realness.

In Table 2, we compare our cManiGAN with GeNeVa (El-Nouby et al. 2019), TIM-GAN (Zhang et al. 2021), ManiGAN (Li et al. 2020a) and TediGAN (Xia et al. 2021a) in terms of FID, IS, image accuracy, and inside-mask accuracy. To better compare these methods for image manipulation, we consider structure-based and attribute-based operations (as mentioned in Sect.), and we divide the evaluation into two types: Type 1 focuses on the “remove/add” operations, and Type 2 considers only “attribute/shape change” operations. These metrics measure the quality of the generated results, reflecting the resulting realness of the synthesized images. From this table, we see that our cManiGAN reported comparable or improved FID and IS scores on both types of operations. It is worth pointing out that, while the best FID scores were reported by TIM-GAN (Zhang et al. 2021) in Type 1 comparisons, they require *ground-truth target images* during training, while others (including ours) do not

have such a requirement.

To further quantify the output accuracy, we report the multi-label classification accuracy on the produced images, and also the single-label classification accuracy on the image regions masked by the ground truth masks. As listed in the above table, we see that cManiGAN achieved either the best or the second-best image accuracy and in-mask accuracy among all methods on two types (i.e, Type 1 as add/remove and Type 2 as attribute/shape change) of operations, which verify that our method successfully edits the image with visual correctness. We observe that GeNeVa tended to “remove” objects in the reference image not following the instruction, resulting in degraded image-level accuracy even with satisfactory in-mask accuracy on Type 1 operations (i.e, add/remove). This is mainly due to the fact that it is optimized on such operations and cannot easily generalized to complex ones like “add”.

Semantic Relevance and Correctness. In Table 2, we report the accuracy of the interpreter and retrieval scores to measure the semantic relevance between the manipulated images and the instructions, as well as the visual correctness of the generated objects. From this table, we see that our cManiGAN again outperformed favorably against baseline methods across both types of operations (i.e., add/remove and attribute/shape change). While GeNeVa and TIMGAN produced visually realistic images, their corresponding interpretation accuracy and retrieval scores were not satisfactory, indicating their lack of ability in comprehending complex instructions to edit the images with semantic and visual correctness. In comparison, our approach consistently achieved higher scores on interpretation accuracy and retrieval scores with remarkable margins. Additional user studies in terms of generating realism and semantic relevance are presented in the supplementary materials.

Real-World Target-Free Images

We now consider the COCO dataset, a challenging real-world dataset with no target image data (i.e., no retrieval scores R@N can be measured). For quantitative comparison purposes, we compare with ASE (Shetty, Fritz, and Schiele 2018), which focuses on object removal in the scene image from a weakly supervised setting, and the results are shown in Table 4(a). From this table, we see that our cManiGAN outperformed ASE in terms of both image quality and semantic correctness. More specifically, our method improved the FID score while the interpretation accuracy was improved by nearly 10%. As for qualitative analysis and more experiments on COCO, please refer to our supplementary materials.

Ablation Studies and Remarks

Design of cManiGAN. Table 4(b) assesses the contributions of each deployed module in our cManiGAN, and thus verifies the design of our proposed model. To justify the two stage design of our generator G , we ablate the localizer L , and utilize only the in-painter P to produce the generated image by observing the reference image and the instruction. One can find that, by adding the localizer, our generator is

	FID ↓ IS ↑		image acc (%)	Inside-mask acc (%)	Interp. acc (%)
Upper bound	-	-	98.96	89.56	67.16
Ours w/o L	228.7	1.11	72.52	24.38	0.667
Ours w/o R (cycle)	68.08	2.07	83.47	41.40	28.27
Ours w/o \mathcal{I}	44.08	2.11	93.73	41.01	35.14
Ours w/o R, \mathcal{I}	77.56	2.08	80.22	39.70	26.55
Ours	39.41	2.22	93.41	41.92	37.46

Table 4: Ablation studies on CLEVR. Note that L , \mathcal{I} and cycle denote the localizer in generator, cross-modal interpreter and operational cycle-consistency losses, respectively.

able to comprehend the complex instruction and locate the target location, which enforces the generator to manipulate within target location only, improving both the semantic correctness of the generated image and the overall visual quality of the generated image. To verify the effectiveness of the interpreter \mathcal{I} during training, we show that adding such a module (comparing the third and fifth rows of this table) into our baseline would further improve the performance, with improved visual realism and semantic correctness.

Finally, we verify the contribution of the operational cycle-consistency and observe improved visual quality scores in FID. This verifies that our operational cycle-consistency serves as a potential pixel-level training feedback without observing the target images. By combining all of the above designs, our full version of cManiGAN achieves the best results in Table 4(b). Thus, the design of our cManiGAN can be successfully verified. For more ablation studies of objective function and reasoner module, please refer to our supplementary materials.

Conclusion

We presented a Cyclic Manipulation GAN (cManiGAN) for target-free text-guided image manipulation, realizing *where* and *how* to edit the input image with the given instruction. To address this task, a number of network modules are introduced in our cManiGAN. The image editor learns to manipulate an image in a two-stage manner, locating the image region of interest followed by completing the output image. In order to guarantee that the output image would exhibit visual and semantic correctness at pixel and world levels, our cManiGAN has unique modules of a cross-modal interpreter and a reasoner, leveraging auxiliary semantics self-supervision. The former associates the image and text modalities and serves as a semantic discriminator, which enforces the authenticity and correctness of the output image via word-level training feedback. The latter is designed to infer the “undo” instruction, allowing us to train cManiGAN with operational cycle-consistency, providing additional pixel-level guidance. With extensive quantitative and qualitative experiments, including user studies, the use of the model is shown to perform favorably against state-of-the-art methods which require different degrees of supervision or be applicable to a limited amount of manipulation tasks.

Acknowledgements

This work is supported in part by the National Science and Technology Council of Taiwan under grant NSTC 111-2634-F-002-020. We also thank the National Center for High-performance Computing (NCHC) and Taiwan Computing Cloud (TWCC) for providing computational and storage resources. Finally, we are especially grateful to Dr. Yunhsuan Sung and Dr. Da-Cheng Juan from Google Research for giving their valuable research advice.

References

- Bird, S.; Klein, E.; and Loper, E. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. ” O’Reilly Media, Inc.”.
- Choi, Y.; Choi, M.; Kim, M.; Ha, J.-W.; Kim, S.; and Choo, J. 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8789–8797.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Dhamo, H.; Farshad, A.; Laina, I.; Navab, N.; Hager, G. D.; Tombari, F.; and Rupperecht, C. 2020. Semantic image manipulation using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5213–5222.
- El-Nouby, A.; Sharma, S.; Schulz, H.; Hjelm, D.; Asri, L. E.; Kahou, S. E.; Bengio, Y.; and Taylor, G. W. 2019. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10304–10312.
- Everingham, M.; Eslami, S. A.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2015. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1): 98–136.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Herzig, R.; Bar, A.; Xu, H.; Chechik, G.; Darrell, T.; and Globerson, A. 2020. Learning canonical representations for scene graph to image generation. In *European Conference on Computer Vision*, 210–227. Springer.
- Iizuka, S.; Simo-Serra, E.; and Ishikawa, H. 2017. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4): 1–14.
- Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, 694–711. Springer.
- Johnson, J.; Gupta, A.; and Fei-Fei, L. 2018. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1219–1228.
- Johnson, J.; Hariharan, B.; Van Der Maaten, L.; Fei-Fei, L.; Lawrence Zitnick, C.; and Girshick, R. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2901–2910.
- Li, B.; Qi, X.; Lukasiewicz, T.; and Torr, P. H. 2019. Controllable text-to-image generation. *arXiv preprint arXiv:1909.07083*.
- Li, B.; Qi, X.; Lukasiewicz, T.; and Torr, P. H. 2020a. Manigan: Text-guided image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7880–7889.
- Li, B.; Qi, X.; Torr, P. H.; and Lukasiewicz, T. 2020b. Lightweight generative adversarial networks for text-guided image manipulation. *arXiv preprint arXiv:2010.12136*.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.
- Liu, X.; Lin, Z.; Zhang, J.; Zhao, H.; Tran, Q.; Wang, X.; and Li, H. 2020. Open-edit: Open-domain image manipulation with open-vocabulary instructions. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, 89–106. Springer.
- Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vllbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*.
- Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J. R.; Bethard, S.; and McClosky, D. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 55–60.
- Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, 2642–2651. PMLR.
- Patashnik, O.; Wu, Z.; Shechtman, E.; Cohen-Or, D.; and Lischinski, D. 2021. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2085–2094.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.
- Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; and Lee, H. 2016. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, 1060–1069. PMLR.
- Shetty, R.; Fritz, M.; and Schiele, B. 2018. Adversarial scene editing: Automatic object removal from weak supervision. *arXiv preprint arXiv:1806.01911*.
- Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

- Viazovetskyi, Y.; Ivashkin, V.; and Kashin, E. 2020. Stylegan2 distillation for feed-forward image manipulation. In *European conference on computer vision*, 170–186. Springer.
- Vo, N.; Jiang, L.; Sun, C.; Murphy, K.; Li, L.-J.; Fei-Fei, L.; and Hays, J. 2019. Composing text and image for image retrieval-an empirical odyssey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6439–6448.
- Wang, H.; Lin, G.; Hoi, S. C.; and Miao, C. 2021. Cycle-consistent inverse GAN for text-to-image synthesis. In *Proceedings of the 29th ACM International Conference on Multimedia*, 630–638.
- Xia, W.; Yang, Y.; Xue, J.-H.; and Wu, B. 2021a. TediGAN: Text-Guided Diverse Face Image Generation and Manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2256–2265.
- Xia, W.; Yang, Y.; Xue, J.-H.; and Wu, B. 2021b. Towards Open-World Text-Guided Face Image Generation and Manipulation. *arXiv preprint arXiv:2104.08910*.
- Yang, C.-F.; Fan, W.-C.; Yang, F.-E.; and Wang, Y.-C. F. 2021. LayoutTransformer: Scene Layout Generation With Conceptual and Spatial Diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3732–3741.
- Zhang, T.; Tseng, H.-Y.; Jiang, L.; Yang, W.; Lee, H.; and Essa, I. 2021. Text as neural operator: Image manipulation by text instruction. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1893–1902.