Memory-Driven Text-to-Image Generation

Bowen Li[1]
bowen.li@cs.ox.ac.uk
Philip H. S. Torr[1]
philip.torr@eng.ox.ac.uk
Thomas Lukasiewicz[2,1]
thomas.lukasiewicz@cs.ox.ac.uk

[1] University of Oxford
Oxford, UK
[2] TU Wien
Vienna, Austria

Abstract

We introduce a memory-driven semi-parametric approach to text-to-image generation, which is based on both parametric and non-parametric techniques. The non-parametric component is a memory bank of image features constructed from a training set of images. The parametric component is a generative adversarial network. Given a new text description at inference time, the memory bank is used to selectively retrieve image features that are provided as basic information of target images, which enables the generator to produce realistic synthetic results. We also incorporate content information into the discriminator, together with semantic features, allowing the discriminator to make a more reliable prediction. Experimental results demonstrate that the proposed memory-driven semi-parametric approach produces realistic images, compared to purely parametric approaches, in terms of both visual fidelity and text-image semantic consistency.

1 Introduction

How to effectively produce realistic images from given natural language descriptions with semantic alignment has drawn much attention, because of its tremendous potential applications in art, design, and video games, to name a few. Recently, with the vast development of generative adversarial networks [7, 8, 33] in realistic image generation, text-to-image generation has made much progress, where the progress has been mainly driven by parametric models — deep networks use their weights to represent all data concerning a realistic appearance [4, 12, 21, 22, 24, 26, 39, 40, 49, 50, 51, 56].

Although these approaches can produce realistic results on well-structured datasets, containing a specific class of objects at the image center with fine-grained descriptions, such as birds [48] and flowers [36], there is still much room to improve. Besides, they usually fail on more complex datasets, which contain multiple objects with diverse backgrounds, e.g., COCO [30]. This is likely because, for COCO, the generation process involves a large variety in objects (e.g., pose, shape, and location), backgrounds, and scenery settings. Thus, it is much easier for these approaches to only produce text-semantic-matched appearances instead of capturing difficult geometric structure. As shown in Fig. 1, current approaches are only capable of producing required appearances semantically matching the given descriptions (e.g., white and black stripes for zebra), but objects are unrealistic with distorted shape.

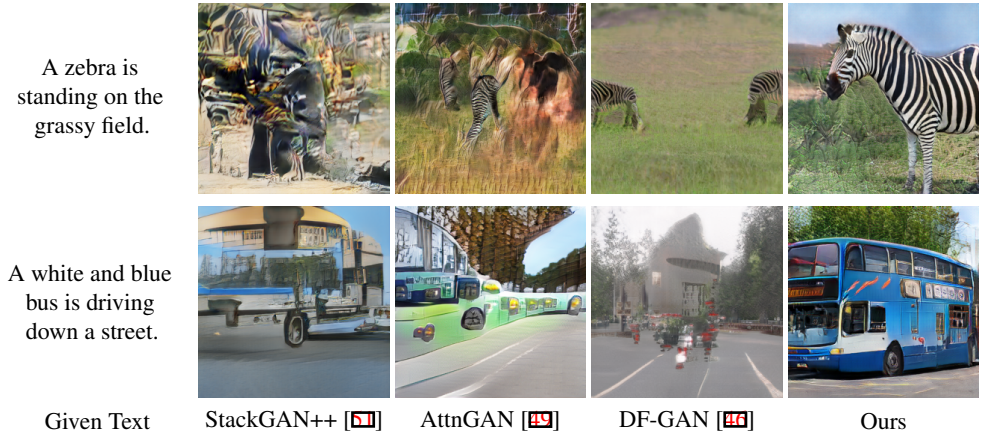

Figure 1: Examples of text-to-image generation on COCO. Current approaches only generate low-quality images with unrealistic objects. In contrast, our method can produce realistic images, in terms of both visual appearances and geometric structure.

Furthermore, these approaches are in contrast to earlier works on image synthesis, which were based on non-parametric techniques that could make use of large datasets of images at inference time [3, 9, 14, 18, 55]. Although parametric approaches can enable the benefits of end-to-end training of highly expressive models, they lose the strength of earlier non-parametric techniques, as they fail to make use of large datasets of images at inference time.

In this paper, we introduce a memory-driven semi-parametric approach to text-to-image generation, where the approach takes the advantages of both parametric and non-parametric techniques. The non-parametric component is a memory bank of disentangled image features constructed from a training set of real images. The parametric component is a generative adversarial network. Given a text description at inference time, the memory bank is used to selectively retrieve compatible image features that are provided as basic information, allowing the generator to directly draw clues of target images, and to produce realistic synthetic results.

Besides, to further improve the differentiation ability of the discriminator, we incorporate content information into it. This is because, to make a prediction, the discriminator usually relies on semantic features, extracted from a given image using a series of convolution operators with local receptive fields. However, when the discriminator goes deeper, less content details are preserved, including exact geometric structure information [6, 15]. We think that the loss of content details is likely one of the reasons why current approaches fail to produce realistic shapes for objects on difficult datasets, such as COCO. Thus, the adoption of content information allows the model to exploit the capability of content details and then improve the discriminator to make the final prediction more reliable.

Finally, an extensive experimental analysis is performed, which demonstrates that our proposed semi-parametric method can generate realistic images from natural language, compared to purely parametric models, in terms of both visual appearances and geometric structure.

2 Related Work

Text-to-image generation has made much progress because of the success of generative adversarial networks (GANs) [8] in realistic image generation. Zhang et al. [50] proposed a multi-stage architecture to generate realistic images progressively. Then, attention-based

methods [22, 49] are proposed to further improve the results. Zhu et al. [56] introduced a dynamic memory module to refine image contents. Besides, extra information is adopted on the text-to-image generation process, such as scene graphs [2, 16] and layout (e.g., bounding boxes or segmentation masks) [12, 13, 27]. However, none of the above approaches adopt non-parametric techniques to make use of large datasets of images at inference time, neither feed content information into the discriminator to enable a finer training feedback. Also, our method does not utilize any additional semantic information, e.g., scene graphs and layout.

Text-guided image manipulation is related to our work, where the task also takes natural language descriptions and real images as inputs, but it aims to modify the images using given texts to achieve semantic consistency [8, 23, 25, 34]. Differently from it, our work focuses mainly on generating novel images, instead of editing some attributes of the given images. Also, the real images in the text-guided image manipulation task behave as a condition, where the synthetic results should reconstruct all text-irrelevant attributes from the given real images. Differently, the real images in our work are mainly to provide the generator with additional cues of target images, in order to ease the whole generation process.

Memory bank. Qi et al. [38] introduced a semi-parametric approach to realistic image generation from semantic layouts. Li et al. [20] used real image features asimage prior to provide clues of target images in image generation. Li et al. [28] used the stored image crops to determine the appearance of objects. Tseng et al. [47] used a differentiable retrieval process to select mutually compatible image patches. Differently, instead of using a concise semantic representation (a scene graph as input), which is less user-friendly and has limited context of general descriptions, we use natural language descriptions as input. Also, Li et al. [29] designed a memory structure to parse the textual content. Differently, our method simply uses a deep network to extract image features, instead of involving complex image preprocessing to build a memory bank.

3 Overview

Given a sentence S, we aim to generate a fake image I', which is semantically aligned with S. Our proposed semi-parametric approach is trained on sets of paired text description and real image features v, denoted by (S, v), where image features are extracted by a pretrained VGG-16 network [45] from real images. These sets are also used to build a memory bank M, so each element in M is an image feature extracted from a training image, associated with corresponding semantically-matched text descriptions from the training datasets.

At inference time, we are given a novel text description S that was not seen during training. Then, S is used to retrieve semantically-aligned image features from the memory bank M, based on designed matching algorithms (more details are shown in Section 3.2). Next, the retrieved image features v, together with word embeddings w, which are encoded from the given text description S, are fed into the generator to synthesize an output image (see Fig. 2). During the generation process, the generator can further selectively choose semantic information from disentangled image features v_D, and fuses them with hidden features to generate realistic images semantically-aligned with S.

3.1 Memory Bank Representation

The memory bank M contains a set of image features v extracted from training images, and each image feature v is associated with several matched text descriptions that are provided

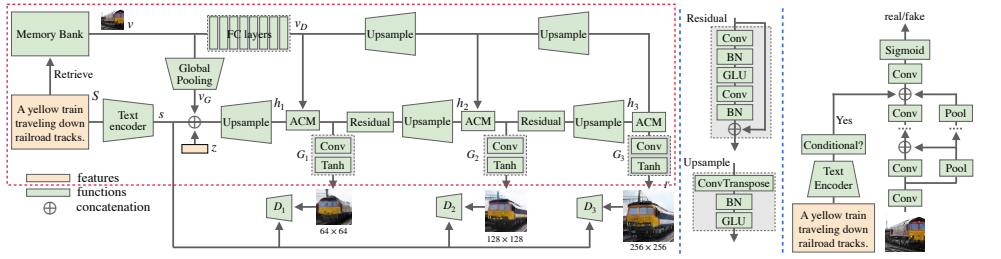

Figure 2: Left: architecture of our proposed method. The red box indicates the inference pipeline that retrieves image features from a memory bank according to the a given description S; during training, we directly feed image features from the text-paired training image. z is a random vector drawn from the Gaussian distribution. Right: architecture of the proposed discriminator with the incorporation of content information.

in the dataset, e.g., in COCO, each image has five matched text descriptions. We also store these matched texts in the memory bank, which are used in text-image matching algorithms, allowing a given unseen text to better retrieve image features at inference time.

3.2 Retrieval

Given a new text description S, to retrieve the most compatible image features from the memory bank M, we design several matching algorithms and also explored the effectiveness of each algorithm. Basically, our retrieval algorithms are based on the calculation of similarity between text features extracted from the given text description and stored text and image features in the memory bank. We explore three different ways to calculate the similarity: (1) matching between the given text and stored texts, (2) matching between the given text and stored image features, and (3) matching between the given text and stored text and image features. In each way, we also explore the effectiveness of different levels of information, where text includes sentence and word levels, and image includes global and regional levels. A detailed description and comparison between algorithms is shown in the supplementary.

4 Memory-Driven Generative Adversarial Networks

To generate high-quality synthetic images from natural language descriptions, we propose to incorporate image features v, along with the given sentence S, into the generator.

4.1 Generator with Image Features

To avoid the identity mapping and also to make full use of image features v in the generator, we first average v on each channel to filter potential content details (e.g., overall spatial structure) contained in v, getting a global image feature v_G, where v_G only keeps basic information of the corresponding real image I, serving as basic image priors. By doing this, the model can effectively avoid copying and pasting from I, and greatly ensure the diversity of output results, especially on the first stage. This is because the following stages focus more on refining basic images produced by the first stage, according to adding more details and improving their resolution, shown in Fig. 2.

However, only feeding the global image feature v_G at the beginning of the network, the model may fail to fully utilize the cues contained in the image features v. Thus, we further incorporate the image features v at each stage of the network. The reason to feed image features v rather than the global feature v_G at the following stages is that v contains more information about the desired output image, such as image contents and geometric structure of objects, where these details can work as candidate information for the main generation pipeline to select. To enable this regional selection effect, we adopt the text-image affine combination module (ACM) [23], which can selectively fuse text-required image information within v into the hidden features h, where h is generated from the given text description S.

Why does the generator with image features work better? Ideally, the generator produces a sample, e.g., an image, from a latent code, and the distribution of these samples should be indistinguishable from the training distribution, where the training distribution is actually drawn from the real samples in the training dataset. Based on this, incorporating image features from real images in the training dataset into the generator allows the generator to directly draw cues of the desired distribution that it eventually needs to generate. Besides, the global feature v_G and disentangled image features v_D can provide basic information of target results in advance, and also work as candidate information, allowing the model to selectively choose text-required information without generating it by the model itself, and thus easing the whole generation process. To some extent, the global feature v_G can be seen as the meta-data of target images, which may contain information about what kinds of objects to generate, e.g., zebra or bus, and v_D is able to provides basic information of objects, e.g., the spatial structure like four legs and one head for the zebra, and the rectangle shape for the bus.

4.2 Discriminator with Content Information

To further improve the discriminator to make a more reliable prediction, relative to both visual appearances and geometric structure, we propose to incorporate content information into it. This is because, in a deep convolutional neural network, when the network goes deeper, the less content details are preserved, including the exact shape of objects [6, 15]. We think the loss of content details may prevent the discriminator to provide fine-grained shape-quality-feedback to the generator, which may cause the difficulty for the generator to produce realistic geometric structure. Also, Zhou et al. [54] showed that the empirical receptive field of a deep convolutional neural network is much smaller than the theoretical one especially on deep layers. So, using convolution operators with a local receptive field only, the network may fail to capture the spatial structure of objects when the size of objects exceeds the receptive field.

To incorporate content details, we propose to generate a series of image content features, $\{a_{128}, a_{64}, a_{32}, \ldots, a_4\}$, by aggregating different image regions via applying pooling operators on the given real or fake features. The size of these content features is from $a_{128} \in \mathbb{R}^{C\times 128\times 128}$ to $a_4 \in \mathbb{R}^{C\times 4\times 4}$, where C represents the number of channels, and the width and the height of the next image content features are $1/2$ the previous one. Thus, the given image is pooled into representations for different regions, from fine- (a_{128}) to coarse-scale (a_4), which can preserve content information of different subregions, such as the spatial structure of objects. Then, these features are concatenated with the corresponding hidden features on the channel-wise direction, incorporating content information into the discriminator.

The number of different-scale content features can be modified, which is dependent on the size of given images. These features aggregate different image subregions by repetitively adopting fixed-size pooling kernels with a small stride. Thus, these content features maintain a reasonable small gap for image information. For the type of pooling operation between max

and average, we perform comparison studies to show the difference in Section 5.2.

Why does the discriminator with content information work better? Basically, the discriminator in a GAN is simply a classifier [8]. It tries to distinguish real data from the data created by the generator (note that in our method, we implement the minmax loss in the loss function, instead of the Wasserstein loss [1]). Also, the implementation of content information has shown its great effectiveness on classification [10, 19, 32, 37] and semantic segmentation [31, 53]. Based on this, incorporating content information into the discriminator is helpful, allowing the discriminator to make a reliable prediction on complex datasets, especially for datasets with complex image scenery settings, such as COCO.

4.3 Training

There are three stages in the model, and each stage has a generator network and a discriminator network. The generator and discriminator are trained alternatively by minimizing the generator loss $\mathcal{L}_G$ and discriminator loss $\mathcal{L}_D$.

4.3.1 Generator Objective

The generator objective for training a generator at stage i contains an unconditional adversarial loss, a conditional adversarial loss, and a text-image matching loss $\mathcal{L}_{\text{DAMSM}}$ [49].

$$\begin{aligned}\mathcal{L}_{G_i} = &\underbrace{-\frac{1}{2}E_{z\sim P_z, v\sim P_{\text{data}}}[\log(D_i(G_i(z,S,v)))]}_{\text{unconditional adversarial loss}} \\ &\underbrace{-\frac{1}{2}E_{z\sim P_z, v\sim P_{\text{data}}}[\log(D_i(G_i(z,S,v),S))]}_{\text{conditional adversarial loss}} + \lambda\mathcal{L}_{\text{DAMSM}},\end{aligned} \tag{1}$$

where G_i and D_i represent the corresponding generator network and discriminator network at stage i, respectively, S is the text description, v is the image features that are extracted from the corresponding real image I that correctly semantically matches S, where I is sampled from the true distribution P_{data}, z is a noise vector drawn from the Gaussian distribution P_z.

Thus, the complete objective function for training the generator networks is:

$$\mathcal{L}_G = \sum_{k=1}^{K}(\mathcal{L}_{G_i}), \tag{2}$$

where K is the total number of stages in the network.

4.3.2 Discriminator Objective

The discriminator objective for training a discriminator at stage i contains an unconditional adversarial loss and a conditional adversarial loss:

$$\begin{aligned}\mathcal{L}_{D_i} = &\underbrace{-\frac{1}{2}E_{I_i\sim P_{\text{data}}}[\log(D_i(I_i))] - \frac{1}{2}E_{z\sim P_z}[\log(1-D_i(G_i(z,S,v)))]}_{\text{unconditional adversarial loss}} \\ &\underbrace{-\frac{1}{2}E_{I_i\sim P_{\text{data}}}[\log(D_i(I_i,S))] - \frac{1}{2}E_{z\sim P_z}[\log(1-D_i(G_i(z,S,v),S))]}_{\text{conditional adversarial loss}},\end{aligned} \tag{3}$$

where I_i denotes the real image sampled from the true image distribution P_{data} at stage i. Thus, the complete objective function for training the discriminator networks is:

$$\mathcal{L}_D = \sum_{k=1}^{K} (\mathcal{L}_{D_i}) + R_1(\psi), \tag{4}$$

where $R_1(\psi)$ is a regularization term described in the paper. This regularization term is derived from zero-centered gradient penalties [43] on local stability, which penalizes the discriminator for deviating from the Nash equilibrium. This ensures that when a GAN-based model converges (i.e., the generator produces the true data distribution), the discriminator cannot create a non-zero gradient orthogonal to the data manifold without suffering a loss in the GAN game.

5 Experiments

To verify the effectiveness of our proposed method in realistic image generation from text descriptions, we conducted extensive experiments on the CUB bird [48] and the more complex COCO [30] dataset, where COCO contains multiple objects with diverse backgrounds.

Evaluation metrics. We adopt the Fréchet inception distance (FID) [11] as the primary metric to quantitatively evaluate the image quality and diversity. Since (compared to the Inception score (IS) [44]) FID is more consistent with human evaluation [17], we also provide IS as a supplementary result. In our experiments, we use 30k synthetic images vs. 30k real test images to calculate the FID and IS values. However, as FID cannot reflect the relevance between an image and a text description, we use the R-precision [49] to measure the correlation between a generated image and its corresponding text. Following [17], we also report SOA-C (i.e., the percentage of images per class in which a desired object is detected) and SOA-I (i.e., the percentage of images in which a desired object is detected).

Implementation. There are three stages in the model, and each stage has a generator network and a discriminator network. The number of stages can be modified, which depends on the resolution of the output image. We utilize a deep neural network layer relu5_3 of a pre-trained VGG-16 to extract image features v, which is able to filter content details in I and keep more semantic information. In the discriminator, the number of different-scale image content features can be modified, which is related to the size of the given image. A same-size pooling kernel with a small stride (stride $= 2$) is repeatedly implemented on the image features, to maximize the preservation of the content information. As for the type of pooling operation, average pooling is adopted. The resolution of synthetic results is 256×256. Our method and its variants are trained on a single Quadro RTX 6000 GPU, using the Adam optimizer [17] with the learning rate 0.0002. We preprocess datasets according to the method used in [49]. **No** attention module is implemented in the whole architecture.

5.1 Comparison with Other Approaches

Quantitative comparison. The results are shown in Table 1. Compared to pure parametric approaches with a similar architecture, our method achieves competitive FID and R-precision scores on both datasets, and even has a better performance than OP-GAN, where OP-GAN adopts bounding boxes. This indicates that (1) our method can produce realistic images from given text descriptions, in terms of image quality and diversity, and (2) synthetic results

Table 1: Quantitative comparison: IS, FID, R-precision, SOA-C, and SOA-I of current methods and our approach on the CUB and COCO datasets. CP-GAN, Obj-GAN, and OP-GAN adopt additional bounding boxes in their methods.

	CUB			COCO				
Method	IS	FID score	R-precison	IS	FID score	R-precison	SOA-C	SOA-I
Real Images	25.34	-	89.17	34.88	-	92.71	74.97	80.84
AttnGAN [49]	4.36	23.98	67.82	25.89	32.32	85.47	25.88	39.01
ControlGAN [27]	4.58	13.92	69.33	24.06	33.58	72.43	-	-
MirrorGAN [40]	4.56	-	57.67	26.47	-	74.52	-	-
DM-GAN [58]	4.75	16.09	72.31	32.32	32.64	88.56	33.44	48.03
DF-GAN [46]	5.10	14.81	-	-	21.42	-	-	-
XMC-GAN [52]	-	-	-	30.45	9.33	-	50.94	71.33
LAFITE	**5.97**	**10.48**	-	32.34	**8.12**	-	61.09	74.78
DALL-E [43]	-	-	-	-	∼20	-	-	-
GLIDE [35]	-	-	-	-	12.89	-	-	-
CP-GAN [29]	-	-	-	**52.73**	55.82	**93.59**	**77.02**	**84.55**
Obj-GAN [27]	-	-	-	30.29	36.52	87.84	27.14	41.24
OP-GAN [10]	-	-	-	27.88	24.70	89.01	35.85	50.47
Ours	5.91	10.49	**73.87**	29.36	19.47	90.32	47.46	65.83

produced by our method are semantically aligned with the given text descriptions. Compared to the large-scale CLIP-based [41] method LAFITE and GLIDE, transformer-based method DALLE, and contrastive-learning-based XMC-GAN, our approach achieves a competitive performance on CUB bird and COCO, reflected by the scores on different evaluation metrics. Although CP-GAN achieves higher IS and SOA scores, both our FID and visual inspection of randomly selected images indicate that our image quality is much higher than CP-GAN's. This may be due to the issue that IS and SOA do not penalize intra-class mode dropping (low diversity within a class) — a model that generates one "perfect" sample for each class can achieve good scores on IS and SOA [12, 27, 52].

Qualitative comparison. In Fig. 3, we present synthetic examples produced by our method at 256×256, along with the corresponding retrieved images that provide image features. As we can see, our method can produce high-quality results on CUB and COCO, with respect to realistic appearances and geometric structure, and also semantically matching the given text descriptions. Besides, the synthetic results are different from the retrieved image features, which indicates that there is no significant copy-and-paste problem in our method.

Diversity evaluation. To further evaluate the diversity of our method, we fix the given text description and the corresponding retrieved image features, and only change the given noise z to generate output images, shown in Fig. 5. When we fix the sentence and image features and only change the noise, our method can generate obviously different images, but they still semantically match the given sentence and also make use of information from the image features. More evaluations are shown in the supplementary material.

Figure 5: Diversity. The top row shows the fixed sentence and image features, where we use the corresponding images to represent image features for a better visualization. The bottom presents diverse synthetic images produced by only changing the input noise z.

Figure 3: Qualitative results on CUB and COCO: in the top row is the given sentences; middle row: the image features extracted from the memory bank M (we use corresponding images to represent the image features for a better visualization); bottom row: the synthetic results.

Figure 4: Qualitative comparison between AttnGAN [49], DF-GAN [46], and Ours on COCO.

Semantic information exploration. Here, we further verify whether our method suffers from a copy-and-paste problem, according to explore whether our method can make use of semantic information contained in the retrieved image features. To verify this, instead of extracting image features from RGB images, we use segmentation masks to provide semantic image features, shown in Fig. 6. As we can see, although there is no content information provided in the given segmentation masks, our method is still able to generate realistic images, which indicates that our method can make use of semantic information contained in the image features, instead of simply copying and pasting the retrieved image features to produce output images.

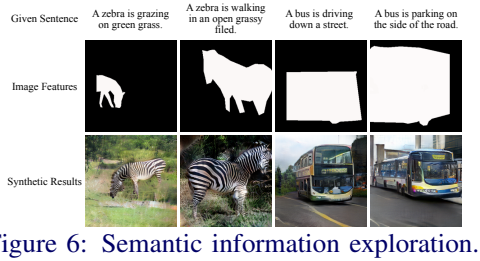

Figure 6: Semantic information exploration. Top row: given sentences; middle row: image features, represented by the corresponding segmentation masks for a better visualization; bottom row: synthetic images.

5.2 Component Analysis

Effectiveness of the image features. To understand the effectiveness of image features in the generator, we conduct an ablation study, shown in Table 2. Without image features, the model "Ours w/o Feature" achieves worse quantitative results on both FID and R-precision compared with the baseline, which verifies their effectiveness on high-quality image generation.

Interestingly, without image features, even our method becomes a pure text-to-image generation method, similar to other baselines, but the FID of "Ours w/o Feature" is still competitive with other baselines. This indicate that even without the image features fed into our method, our method can still generate better synthetic results, with respect to image quality and diversity. We think that this is mainly because with the help of content information, our better discriminator is able to make a more reliable prediction on complex datasets, which in turn encourages the generator to produce better synthetic images.

Table 2: Ablation studies: "Ours w/o Feature" denotes without feeding image features into the generator, "Ours w/o Disen." denotes without using the fully connected layers to disentangle image features v, "Ours w/o Disen.*" is for mismatched pairs, "Ours w/o Content" denotes without incorporating content information into the discriminator, "Ours w/o Reg." denotes without using the regularization in the discriminator, "Ours w/ Max" denotes using maximum pooling to extract content information, and "Ours w/ Aver" denotes using average pooling.

Method	FID	R-psr
Ours w/o Feature	22.20	84.63
Ours w/o Disen.	18.82	92.17
Ours w/o Disen.*	18.80	67.05
Ours w/o Content	20.96	88.95
Ours w/o Reg.	27.12	82.97
Ours w/ Max	26.12	83.11
Ours w/ Aver (Full Model)	**19.47**	**90.32**

Effectiveness of the disentanglement. Here, we show the effectiveness of the fully connected layers applied on the image features v. Interestingly, from Table 2, the "model w/o Disen." achieves better FID and R-precision compared with the baseline. This is likely because the model may suffer from an identity mapping problem. To verify this identity mapping problem, we conduct another experiment, where we feed the mismatched sentence and image pairs into the network without using search algorithms, denoted "model w/o Disen.*". As we can see, on mismatched pairs, although FID is still low, the R-precision degrades.

Content information. In Table 2, FID and R-precision degrade when the discriminator does not adopt the content information. This may indicate that content information can effectively strengthen the differentiation abilities of the discriminator. Then, the improved discriminator can provide the generator with fine-grained training feedback, regarding geometric structure, thus facilitating training a better generator to produce higher-quality synthetic results.

Comparison between different pooling types. In Table 2, as we can see, the model with average pooling works better than max pooling. We think that this is likely because max pooling fails to capture contextual information between neighboring pixels, because it only picks the maximum value among a region of pixels, while average pooling calculates the average value between them.

Effectiveness of the regularization. From Table 2, the model without the regularization has worse quantitative results, compared with the full model. This is because the regularization effectively improves GAN convergence by preventing the generator from training on junk feedback, once the discriminator cannot easily tell the difference between real and fake.

6 Conclusion

We have introduced a memory-driven semi-parametric approach to text-to-image generation, which utilizes large datasets of images at inference time. Also, an alternative architecture is proposed for both the generator and the discriminator. Extensive experimental results on two datasets demonstrate the effectiveness of feeding retrieved image features into the generator and incorporating content information into the discriminator.

Acknowledgments

This work was supported by the UKRI Turing AI Fellowship EP/W002981/1 and the EPSRC/MURI grant EP/N019474/1. We also thank the Royal Academy of Engineering and FiveAI. This work was also supported by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, and by the EPSRC grant EP/R013667/1. We also acknowledge the use of the EPSRC-funded Tier 2 facility JADE (EP/P020275/1) and GPU computing support by Scan Computers International Ltd.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Oron Ashual and Lior Wolf. Specifying object attributes and relations in interactive scene generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4561–4569, 2019.

[3] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2Photo: Internet image montage. *ACM Transactions on Graphics (TOG)*, 28(5):1–10, 2009.

[4] Jun Cheng, Fuxiang Wu, Yanling Tian, Lei Wang, and Dapeng Tao. RiFeGAN rich feature generation for text-to-image synthesis from prior knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10911–10920, 2020.

[5] Hao Dong, Simiao Yu, Chao Wu, and Yike Guo. Semantic image synthesis via adversarial learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5706–5714, 2017.

[6] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.

[7] Jon Gauthier. Conditional generative adversarial networks for convolutional face generation. *Technical report*, page 3, 2015.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[9] James Hays and Alexei A. Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (ToG)*, 26(3):4–es, 2007.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, 2015.

[11] 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. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017.

[12] Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Semantic object accuracy for generative text-to-image synthesis. *arXiv preprint arXiv:1910.13321*, 2019.

[13] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994, 2018.

[14] Phillip Isola and Ce Liu. Scene collaging: Analysis and synthesis of natural images with semantic layers. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3048–3055, 2013.

[15] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[16] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1219–1228, 2018.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[18] Jean-François Lalonde, Derek Hoiem, Alexei A. Efros, Carsten Rother, John Winn, and Antonio Criminisi. Photo clip art. *ACM Transactions on Graphics (TOG)*, 26(3):3–es, 2007.

[19] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178. IEEE, 2006.

[20] Bowen Li and Thomas Lukasiewicz. Lightweight long-range generative adversarial networks. *arXiv preprint arXiv:2209.03793*, 2022.

[21] Bowen Li and Thomas Lukasiewicz. Word-level fine-grained story visualization. *arXiv preprint arXiv:2208.02341*, 2022.

[22] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. Controllable text-to-image generation. In *Advances in Neural Information Processing Systems*, pages 2063–2073, 2019.

[23] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. ManiGAN: Text-guided image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7880–7889, 2020.

[24] Bowen Li, Xiaojuan Qi, Philip Torr, and Thomas Lukasiewicz. Image-to-image translation with text guidance. *arXiv preprint arXiv:2002.05235*, 2020.

[25] Bowen Li, Xiaojuan Qi, Philip Torr, and Thomas Lukasiewicz. Lightweight generative adversarial networks for text-guided image manipulation. *Advances in Neural Information Processing Systems*, 33:22020–22031, 2020.

[26] Bowen Li, Philip HS Torr, and Thomas Lukasiewicz. Clustering generative adversarial networks for story visualization. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 769–778, 2022.

[27] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12174–12182, 2019.

[28] Yikang Li, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. PasteGAN: A semi-parametric method to generate image from scene graph. *Advances in Neural Information Processing Systems*, 32:3948–3958, 2019.

[29] Jiadong Liang, Wenjie Pei, and Feng Lu. CPGAN: content-parsing generative adversarial networks for text-to-image synthesis. In *European Conference on Computer Vision*, pages 491–508. Springer, 2020.

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[31] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. ParseNet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.

[32] Louis Mahon, Eleonora Giunchiglia, Bowen Li, and Thomas Lukasiewicz. Knowledge graph extraction from videos. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 25–32. IEEE, 2020.

[33] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[34] Seonghyeon Nam, Yunji Kim, and Seon Joo Kim. Text-adaptive generative adversarial networks: Manipulating images with natural language. In *Advances in Neural Information Processing Systems*, pages 42–51, 2018.

[35] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[36] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[37] Sai Vidyaranya Nuthalapati, Ramraj Chandradevan, Eleonora Giunchiglia, Bowen Li, Maxime Kayser, Thomas Lukasiewicz, and Carl Yang. Lightweight visual question answering using scene graphs. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3353–3357, 2021.

[38] Xiaojuan Qi, Qifeng Chen, Jiaya Jia, and Vladlen Koltun. Semi-parametric image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8808–8816, 2018.

[39] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Learn, imagine and create: Text-to-image generation from prior knowledge. *Advances in Neural Information Processing Systems*, 32:887–897, 2019.

[40] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019.

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021*, volume 139, pages 8748–8763, 2021.

[42] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[43] Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. *arXiv preprint arXiv:1711.09404*, 2017.

[44] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.

[45] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[46] Ming Tao, Hao Tang, Songsong Wu, Nicu Sebe, Xiao-Yuan Jing, Fei Wu, and Bingkun Bao. DF-GAN: Deep fusion generative adversarial networks for text-to-image synthesis. *arXiv preprint arXiv:2008.05865*, 2020.

[47] Hung-Yu Tseng, Hsin-Ying Lee, Lu Jiang, Ming-Hsuan Yang, and Weilong Yang. RetrieveGAN: Image synthesis via differentiable patch retrieval. In *European Conference on Computer Vision*, pages 242–257. Springer, 2020.

[48] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 dataset. 2011.

[49] 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.

[50] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5907–5915, 2017.

[51] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. StackGAN++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1947–1962, 2018.

[52] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 833–842, 2021.

[53] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2881–2890, 2017.

[54] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene CNNs. *arXiv preprint arXiv:1412.6856*, 2014.

[55] Jun-Yan Zhu, Philipp Krahenbuhl, Eli Shechtman, and Alexei A. Efros. Learning a discriminative model for the perception of realism in composite images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3943–3951, 2015.

[56] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. DM-GAN: Dynamic memory generative adversarial networks for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5802–5810, 2019.