Multi-Modal Deep Learning for Computer Vision and Its Application



Bowen Li

St Catherine's College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2022

To my parents for their love and support.
感谢我的父母给予我的爱和支持。

Acknowledgements

I am most grateful to my advisors, Thomas Lukasiewicz and Philip H. S. Torr, for their ongoing guidance and support. They always provided useful advice and perspective, forming me as a researcher. I am also very grateful to Xiaojuan Qi for her guidance at the beginning of my DPhil.

I am grateful to my DPhil examiners, Ronald Clark and Fabio Cuzzolin, for their careful examination and helpful comments on this thesis.

I also want to thank the Graduate Studies Administrators, Sarah Retz-Jones and Jodie Mattioli, for their ongoing helpful advice.

I am also very grateful to St Catherine's College for becoming my home away from home, for the wonderful friends I made in college, and for the support and care I received from all the college staff.

This work has been supported by an AXA Chair Fund Scholarship.

Lastly, my gratitude goes to my parents, who always believed in me, encouraged me to persevere, and supported me unconditionally, and to all my friends who have been there for me throughout this journey.

Abstract

Although the exponential growth of visual data in various forms, such as images and videos, enables unprecedented opportunities for us to interpret the surrounding environment, natural language is still the main way to convey knowledge and information among us. Therefore, there is currently an increasing demand for building a framework to achieve interaction between pieces of information from different modalities.

In this thesis, I investigate three directions to achieve an effective interaction between multi-modal information. The first direction focuses on building a consistent representation for information with the similar semantic meaning. More specifically, in a high-dimensional semantic space, the representations for similar information should be close to each other within an appropriate range, regardless of their modalities. The second direction is to achieve an effective correlation between image visual attributes with corresponding semantic words, which first requires the network to recognize different semantic information from both the image and text, and then allows it to interact with them. The third direction is to construct a lightweight architecture for models having inputs from multiple domains. This is because when a network involves multi-modal information, the number of trainable parameters might need to considerably increase, which aims to allow the network to comprehensively learn to capture the correlations between the information with a domain gap. The requirements for having significant computing resources might greatly hinder the deployment of a framework, which is not practical for the implementation in real-world applications. The contributions in these directions are as follows.

First, to have a consistent representation, both contrastive and clustering learning are adopted in the generative network, where contrastive learning can maximize the mutual information between paired instances provided by a given dataset, and clustering learning can group instances with similar semantic meaning into the same cluster, and push dissimilar ones far away from each other. By doing this, a structured joint semantic space can be built, where instances with similar semantic meaning can be closely grouped together within an appropriate range to ensure a consistent representation, regardless of their modalities.

Second, to achieve an effective correlation between multi-modal information, three different approaches are proposed, where they effectively correlate image visual attributes with corresponding semantic text descriptions, allowing the network to learn to understand semantic meaning of both text and image information, and then achieving an effective interaction. More specially, for the exploration of correlation, I first

investigate a word-level attention-based connection, helped with a complementary word-level discriminator, where attention allows the network to learn to identify specific image visual attributes, aligned with the corresponding semantic words, and the complementary word-level discriminator provides fine-grained training feedback to allow the network to correctly capture this correlation. Then, a text-image affine combination is introduced, which adopts an affine transformation to combine both text and image features in the generation process, allowing the network to have a regional selection effect, which selectively fuse text-required image attributes into the generation pipeline and preserve text-irrelevant contents. Moreover, a semi-parametric memory-driven approach is proposed, which takes the advantages of both parametric and non-parametric techniques. The non-parametric component is a memory bank of stored pre-processed information constructed from the training dataset, and the parametric component is a neural network. By doing this, the parametric approaches can enable the benefits of end-to-end training of highly expressive models, and the non-parametric techniques allow the network to make full use of large datasets of data at inference time.

Third, two solutions are presented to relieve the cost of computing resources required by a network having inputs from different modalities, allowing the network to be easily implemented in various areas. More specifically, we improve the capabilities of both generator and discriminator in conditional GANs to avoid blindly increasing the number of trainable parameters of a network, and construct a single-directional discriminator by combining two training goals (i.e., having better image quality and text-image semantic alignment) into a single direction (i.e., improving the quality of fusion features) to reduce redundancies in conditional GANs.

This work paves the way for building a lightweight framework to achieve an effective interaction between multi-modal information, and also to be easily deployed in various real-world applications.

Contents

1 Introduction **1**

1.1 The Importance of Interaction between Multi-Modal Information . . . 2

1.2 Research Questions and Contributions . . . 4

1.3 Outline . . . 6

2 Background on Multi-Modal Generative Methods **8**

2.1 Types of Generative Model Architecture . . . 8

2.1.1 Autoencoder . . . 8

2.1.2 Variational Autoencoder . . . 9

2.1.3 Generative Adversarial Networks . . . 11

2.1.4 Conditional Generative Adversarial Networks . . . 11

2.1.5 Multi-Stage Architecture . . . 12

2.2 Attention . . . 12

2.3 Contrastive Learning in GANs and Clustering Learning . . . 13

2.4 Memory Bank . . . 14

2.5 Types of Tasks . . . 15

2.5.1 Text-to-Image Generation . . . 15

2.5.2 Story Visualization . . . 17

2.5.3 Text-Guided Image Manipulation . . . 18

2.6 Types of Datasets . . . 19

2.6.1 CUB Bird . . . 19

2.6.2 MS COCO . . . 19

2.6.3 Pororo-SV . . . 20

2.6.4 Abstract Scenes . . . 20

2.6.5 CLEVR-SV . . . 20

2.7 Evaluation Metrics . . . 21

2.7.1 Inception Score . . . 21

2.7.2 Fréchet Inception Distance . . . 21

2.7.3 Fréchet Story Distance . . . 22
2.7.4 R-Precision . . . 22

3 Consistent Representations for Multi-Modal Information 23
3.1 Background of Story Visualization . . . 23
3.2 Contrastive- and Clustering-Based Approach for the Consistent Representation Construction . . . 24
3.2.1 Model Overview . . . 24
3.2.2 Contrastive Learning for Story Visualization . . . 25
3.2.3 Clustering Learning for Story Visualization . . . 27
3.2.3.1 Fusion Features . . . 28
3.2.3.2 Online K-Means Clustering . . . 28
3.2.3.3 Offline K-Means Clustering . . . 29
3.2.4 Another Components . . . 29
3.2.4.1 Discriminator . . . 30
3.2.4.2 Generator . . . 31
3.2.5 Summary . . . 31
3.3 Experiments . . . 32
3.3.1 Implementation . . . 32
3.3.2 Quantitative Evaluation . . . 33
3.3.3 Qualitative Evaluation . . . 33
3.3.4 Ablation Study . . . 34
3.3.4.1 Individual Contrastive and Clustering Losses . . . 34
3.3.4.2 Combined Contrastive and Clustering Losses . . . 35
3.3.4.3 Different Choices of Contrastive Methods . . . 35
3.3.4.4 Different Choices of Clustering Methods . . . 36
3.3.4.5 Number of Clusters . . . 36
3.3.4.6 Visualization of Effects of Clustering Learning . . . 37
3.3.4.7 Bi-Directional Attention . . . 37
3.3.5 Visualization of Bi-Directional Attention . . . 38
3.3.6 Human Evaluation . . . 39
3.4 Conclusion . . . 39

4 Build Connections between Different-Modal Information 40
4.1 Word-Level Attention-Based Connection with a Complementary Discriminator . . . 40
4.1.1 Background of Controllability in Text-to-Image Generation . . 41
4.1.2 Attention-Based Controllable Text-to-Image Generation 42
4.1.2.1 Attention-Based Connection . . . 42
4.1.2.2 Word-Level Discriminator . . . 44
4.1.2.3 Perceptual Loss . . . 45
4.1.2.4 Objective Functions . . . 46
4.1.2.5 Summary . . . 47
4.1.3 Experiments . . . 47
4.1.3.1 Implementation . . . 47
4.1.3.2 Comparison with State of the Art . . . 48
4.1.3.3 Component Analysis . . . 50
4.1.4 Conclusion . . . 52
4.2 Text-Image Affine Combination . . . 53
4.2.1 Background of Text-Guided Image Manipulation . . . 53
4.2.2 Affine Transformation Based Text-Guided Image Manipulation 54
4.2.2.1 Architecture . . . 54
4.2.2.2 Text-Image Affine Combination Module . . . 56
4.2.2.3 Detail Correction Module . . . 58
4.2.2.4 Training . . . 59
4.2.3 Experiments . . . 61
4.2.3.1 Comparison with State-of-the-Art Approaches . . . 63
4.2.3.2 Ablation Studies . . . 64
4.2.4 Conclusion . . . 66
4.3 Memory-Driven Approach . . . 67
4.3.1 Background of Semi-Parametric Method in Text-to-Image Generation . . . 67
4.3.2 Memory-Driven Generative Adversarial Networks . . . 68
4.3.2.1 Overview . . . 68
4.3.2.2 Generator with Image Features . . . 70
4.3.2.3 Discriminator with Content Information . . . 71
4.3.2.4 Training . . . 72
4.3.3 Experiments . . . 73
4.3.3.1 Comparison with Other Approaches . . . 74

4.3.3.2 Component Analysis 77
4.4 Conclusion 78

5 Lightweight Architecture for Multi-Modal Generative Model 80
5.1 Improving Capabilities of the Network 80
5.1.1 Background of Lightweight Architecture in Text-Guided Image Manipulation 81
5.1.2 Lightweight Generative Adversarial Networks for Image Manipulation 83
5.1.2.1 Word-Level Discriminator 83
5.1.2.2 Word Labelling 83
5.1.2.3 Discriminator Providing Word-Level Training Feedback 84
5.1.2.4 Generator of the Lightweight Architecture 85
5.1.2.5 Objective Functions 86
5.1.3 Experiments 87
5.1.3.1 Comparison with State of the Art 88
5.1.3.2 Component Analysis 90
5.1.4 Conclusion 91
5.2 Single-Direction Discriminator 93
5.2.1 Discriminator with Fusion Features 93
5.2.2 Another Components 96
5.2.2.1 New Sentence Representation 96
5.2.2.2 Extended Word-Level Spatial Attention 97
5.2.3 Experiments 97
5.2.3.1 Implementation 98
5.2.3.2 Quantitative Evaluation 98
5.2.3.3 Qualitative Evaluation 99
5.2.3.4 Component Analysis 100
5.2.3.5 Visualization 102
5.2.3.6 General Applicability of New Sentence Representation 103
5.2.3.7 Human Evaluation 103
5.2.4 Conclusion 104

6 General Conclusion and Perspective 106

Bibliography 109

List of Figures

1.1 Two directions of visual intelligence. 2

2.1 Architecture of the autoencoder. 9
2.2 Architecture of the variational autoencoder. 11
2.3 Illustration of Pororo-SV, CLEVR-SV, and Abstract Scenes. 20

3.1 Illustration of the proposed framework. 25
3.2 Comparison between WSA and BWSA. 30
3.3 Qualitative comparison between methods on the Pororo-SV. 33
3.4 Qualitative comparison between methods on the Abstract Scenes. . . 34
3.5 t-SNE plots of effects of clustering learning. 37
3.6 Heat map of bi-directional attention. 38

4.1 Examples of modifying synthetic images using natural language descriptions. 41
4.2 Architecture of the proposed ControlGAN. 42
4.3 Architecture of the proposed channel-wise attention module and word-level discriminator 43
4.4 Qualitative results on the CUB and COCO. 48
4.5 Qualitative comparison of methods on the CUB and COCO. 49
4.6 Visualization of feature channels and spatial attention. 50
4.7 Ablation studies of channel-wise attention and word-level discriminator. 51
4.8 Effectiveness of the perceptual loss and word-level discriminator. . . . 51
4.9 Examples of text-guided image manipulation. 55
4.10 Architecture of ManiGAN. 56
4.11 Architecture of the text-image affine combination module and the detail correction module. 57
4.12 Ablation studies of the learned W and b. 58
4.13 Plots of different evaluation metrics at different epochs on the CUB and COCO. 62

4.14 Qualitative comparison of methods on the CUB and COCO. 63
4.15 Ablation studies. 64
4.16 Examples of text-to-image generation on the COCO. 69
4.17 Architecture of the network. 70
4.18 Qualitative results on the CUB and COCO. 75
4.19 Qualitative comparison on the COCO. 75
4.20 Diversity. 76
4.21 Semantic information exploration. 77

5.1 Examples of image manipulation using natural language descriptions. 81
5.2 Architecture of the network. 82
5.3 Qualitative comparison of two methods on the CUB. 87
5.4 Qualitative comparison of two methods on the COCO. 88
5.5 Plots of FID over numbers of parameters. 90
5.6 Ablation and comparison studies. 91
5.7 Visualization of attention maps. 92
5.8 Illustration of the normal and proposed discriminators. 93
5.9 Illustration of the training goal of the normal and proposed discriminators. 94
5.10 Architecture of the network. 96
5.11 Comparison between different methods on the Pororo-SV. 99
5.12 Comparison between different methods on the Abstract Scenes. . . . 100
5.13 Visualization of the proposed extended spatial attention and new sentence representation. 102

List of Tables

3.1 Quantitative evaluation between different methods on the Pororo-SV and Abstract Scenes. 32
3.2 Ablation results with different contrastive losses and clustering losses on the Pororo-SV. 35
3.3 Comparison between different choices on the Pororo-SV. 36
3.4 Human evaluation on the Pororo-SV based on three criteria. 39

4.1 Quantitative comparison of baselines and ControlGAN on the CUB and COCO. 47
4.2 Quantitative comparison of state-of-the-art approaches and ManiGAN on the CUB and COCO. 61
4.3 Quantitative comparison of current methods and our approach on the CUB and COCO. 74
4.4 Ablation studies of different components introduced in the proposed memory-driven approach. 78

5.1 Quantitative comparison of the state of the art and our method on the CUB and COCO. 87
5.2 Quantitative comparison of the state of the art and our method on the CUB and COCO. 88
5.3 Quantitative evaluation between different methods on the Pororo-SV and Abstract Scenes. 98
5.4 Component Analysis on the Pororo-SV. 101
5.5 Effects of our new sentence representation adopted in story visualization on the Pororo-SV and text-to-image generation on the CUB. 103
5.6 Results of a human evaluation between Ours and DUCO on the Pororo-SV. 104

List of Publication

Li et al. (2019a): **B. Li**, X. Qi, T. Lukasiewicz, and P. Torr. Controllable text-to-image generation. Advances in Neural Information Processing Systems, pages 2063–2073, 2019. Discussed in Chapter 4.

Li et al. (2020a): **B. Li**, X. Qi, T. Lukasiewicz, and P. Torr. ManiGAN: Text-guided image manipulation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 7880–7889, 2020. Discussed in Chapter 4.

Li et al. (2020c): **B. Li**, X. Qi, P. Torr, and T. Lukasiewicz. Lightweight generative adversarial networks for text-guided image manipulation. Advances in Neural Information Processing Systems, 33:22020–22031, 2020. Chapter 5.

Li et al. (2022a): **B. Li**, P. Torr, and T. Lukasiewicz. Clustering Generative Adversarial Networks for Story Visualization. In Proceedings of the 30th ACM International Conference on Multimedia 2022. Discussed in Chapter 3.

Li and Lukasiewicz (2022b): **B. Li** and T. Lukasiewicz. Word-Level Fine-Grained Story Visualization. In Proceedings of the European Conference on Computer Vision 2022. Discussed in Chapter 5.

Li and Lukasiewicz (2022a): **B. Li** and T. Lukasiewicz. Learning to Model Multimodal Semantic Alignment for Story Visualization. In Findings of Empirical Methods in Natural Language Processing 2022.

Li et al. (2020b): **B. Li**, P. Torr, and T. Lukasiewicz. Image-to-image translation with text guidance. In Proceedings of The British Machine Vision Conference 2022.

Li et al. (2022b): **B. Li**, P. Torr, and T. Lukasiewicz. Memory-Driven Text-to-Image Generation. In Proceedings of The British Machine Vision Conference 2022. Discussed in Chapter 4.

Nuthalapati et al. (2021): S. V. Nuthalapati, R. Chandradevan, E. Giunchiglia, **B. Li**, M. Kayser, T. Lukasiewicz, and C. Yang. Lightweight Visual Question Answering Using Scene Graphs. In Proceedings of the 30th ACM International Conference on Information and Knowledge Management, pages 3353–3357, 2021.

Mahon et al. (2020): L. Mahon, E. Giunchiglia, **B. Li**, and T. Lukasiewicz. Knowledge graph extraction from videos. In 2020 19th IEEE International Conference on Machine Learning and Applications, pages 25–32. IEEE, 2020

Chapter 1

Introduction

Visual perception is one of the most important ability for human beings to interpret the surrounding environment. Every day, the amount of visual information consumed by humans is incredible, as they watch videos, take photographs, enjoy paintings, and share on social media. For example, almost 5 billion videos are watched on YouTube every single day; over 95 million photos are uploaded each day on Instagram only.

The availability of such a large amount of visual data has provided researchers with unprecedented opportunities to build various visual interpreting and synthesis methods like (1) object/scene classification (He et al., 2016; Simonyan and Zisserman, 2014; Szegedy et al., 2015, 2016; Zhou et al., 2014b), (2) object detection (Girshick et al., 2014; Long et al., 2015; Ren et al., 2015; Ronneberger et al., 2015; Howard et al., 2017; Redmon et al., 2016), (3) image captioning (Donahue et al., 2015; Xu et al., 2015), (4) visual question answering (Andreas et al., 2016; Johnson et al., 2017; Lu et al., 2016; Nam et al., 2017; Antol et al., 2015; Anderson et al., 2018; Yang et al., 2016), (5) text-to-image generation (Reed et al., 2016b; Xu et al., 2018; Zhang et al., 2017a, 2018a), (6) image-to-image translation (Isola et al., 2017; Park et al., 2019; Wang et al., 2018; Zhu et al., 2017), (7) text-guided image manipulation (Dong et al., 2017; Li et al., 2020a,c; Nam et al., 2018), (8) story visualization (Li et al., 2019c; Song et al., 2020; Maharana et al., 2021; Maharana and Bansal, 2021), and so on, which aim to achieve a cross-domain transformation between multi-modal representations, e.g., transfer from non-visual data, such as natural language descriptions or scene graphs, into visual information, such as videos or images, or vice verse, and aim to achieve an effective interaction between them.

In this thesis, I focus on exploring the principles to achieve an effective interaction between multi-modal information (e.g., language and visual information), including (1) how to build consistent representations for this information in a joint semantic space, (2) how to effectively connect and also fuse the features from different modalities

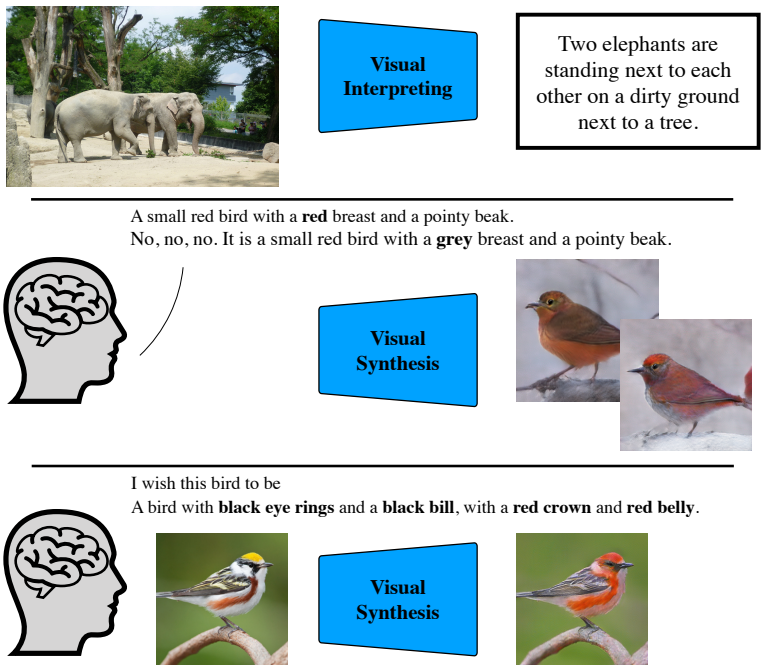


Figure 1.1: Two directions of visual intelligence. (1) Visual interpreting (top) produces natural language descriptions from visual representations; (2) visual synthesis (a) achieves a visual generation from information of a non-visual modal (e.g., natural language and ontology representations) (middle), and (b) manipulates given images using natural language descriptions (bottom).

together, and (3) how to mitigate computing requirements to achieve an effective interaction. My research can enable numerous potential applications in many areas, including design, video games, art, architecture, and medical diagnosis, to name a few.

1.1 The Importance of Interaction between Multi-Modal Information

Indeed, we receive a large amount of visual information all the time via various visual forms, such as videos, photographs, and paintings, to name a few, but only few of us can effectively describe this information visually. Such a huge contrast between pale oral expression and vivid visual perception vastly hinders an effective information transfer between us. However, exploring the principles to achieve an effective interaction between multi-modal features can effectively bridge the communication gap between human and large amount of available visual information, and can remedy numerous difficulties, for example, considering a simple scenario presented in Fig. 1.1, a user wants to describe an adorable bird to her friends that she saw yesterday. Indeed, it is

hard to reproduce a visual image in our mind from the bald description "*a small red bird with a red breast and a pointy beak*", and it could be even hard when she changes her description "*No, no, no. It is a small red bird with a gray breast and a pointy beak*". If the user is an artist, then she can draw the vivid bird efficiently in minutes so that it is almost same as the one she saw; or the user can use image editing tools like Photoshop to transfer an image into the desired one. However, the truth is that most of us are neither artists nor professional tool-users.

Besides, at some professional domains, a given visual information might be hard to understand and recognize for someone who do not have corresponding professional knowledge, while converting this sophisticated visual information into readable and understandable text information can allow laymen to at least roughly understand the meaning of the given visual information. For example, given a X-ray or CT report, without professional medical knowledge, it may be hard for us to completely understand the meaning of it. Moreover, reading a visual information is inefficient and also easy to miss some tiny details, as it takes much time and effort to observe all regional details of it, e.g., for a skilled doctor, he or she can only check limited number of visual reports in a day, and might fail to observe all tiny malignant appearances shown in the report.

Furthermore, there is no doubt that using multiple multi-modal information can comprehensively convey knowledge and information between us, as different single-modal information has its own drawbacks. First, although visual information, such as images and videos, is concrete, vivid, and attractive, it might not be easy to propagate physically, and even for the digital version, it might require several kilobytes per image for storage and bandwidth to transfer. However, for the text, it is concise with specific semantics and syntax, which means that it can be widely propagated and understood following some specific fixed rules, but it might not fully convey the information that a user want to express, as we usually need to imagine some parts of the information in our mind. So, the ability to integrate information from multiple modalities, such as text, image, video, and audio, is essential for effective communication and knowledge propagation. This is particularly important in the age of digital media, where we are constantly bombarded with information from various sources and in different formats. Moreover, the ability to transfer knowledge and information from one domain to another is also crucial for enhancing our understanding of complex concepts and phenomena. For example, by leveraging insights from multiple domains, such as biology, chemistry, and physics, we can gain a more comprehensive understanding of the underlying mechanisms that govern natural phenomena. Therefore, the integration

of information from multiple modalities and the transfer of knowledge between domains are key features of advanced AI systems, that enable us to tackle complex problems and make significant progress in various fields.

Based on above situations, it is better to have a framework that allows us to achieve an effective interaction between multi-modal information. Based on this framework, for example, we can easily generate text descriptions or visual images from the other domain representations (see Fig. 1.1 top and middle), or modify images using natural language descriptions (see Fig. 1.1 bottom). More specifically, this research, for instance, can be applied in different real-world applications:

(1) as for text-to-image generation, it allows non-artists to easily generate desired images by simply providing natural language descriptions (Xu et al., 2018), where the descriptions contain the requirements for desired appearances of objects and background that synthetic images should have. Besides, instead of only generating a single image, it can produce a large amount of diversified images semantically matching a given text description to meet users' different preferences.

(2) as for image-to-text translation, it can help blind people to fully perceive the surrounding environment by effectively converting surrounding visual information into natural language (Donahue et al., 2015; Xu et al., 2015), and also help doctors to quickly diagnose diseases, according to converting visual reports into text explanations (Litjens et al., 2017; Stoyanov et al., 2018; De Bruijne, 2016).

(3) as for image/video editing using natural language descriptions, it can enable a more general and user-friendly image editing tool by simply using natural language descriptions. In particular, it allows to semantically edit parts of an image according to the given text provided by a user, while preserving other contents that are not described in the text (Li et al., 2020a). Moreover, this research can even achieve automatic video editing by using text descriptions, for instance, users are able to add/remove/modify different objects by simply providing a natural language description without going through all frames and editing them manually (Fried et al., 2019).

Given the above-mentioned usages of interaction between multi-modal information, there is currently an increasing demand for a framework to achieve it, which has been the motivation to dedicate my thesis to this topic.

1.2 Research Questions and Contributions

To achieve an effective interaction between multi-modal information, first, it is important to have a consistent representation for information with the similar semantic

meaning. This is because information from different domains may have their own representations, and thus, in a high-dimensional semantic space, the locations of these features may be far away from each other, even though they express the similar semantic meaning. This also happens for information that are from the same domain (Mikolov et al., 2013), e.g., for two sentences with the same meaning but using synonym, such as "cat" and "kitty", the representations for both might not be exactly same in a semantic space. This situation may be even worse when the dimension of features increases, caused by the curse of dimensionality (Bellman, 1966): as the number of features or dimensions grows, the amount of data we need to generalize accurately grows exponentially. So, one question arises: *how to build a consistent representation for information from the same or different domains with the similar semantic meaning?* This is important, as having a consistent representation means that information with the similar semantic meaning can have fixed and close locations in the high-dimensional semantic space, and thus even this information from different modalities can still be grouped together to enable an easy transformation from one to the other within an appropriate range in the space, which is aligned with the goal of this thesis to achieve an effective interaction between multi-modal information.

To consider similarity between instances, we first seek help from *contrastive learning*, where contrastive learning is a powerful scheme for self-supervised representation learning (Oord et al., 2018; He et al., 2020; Chen et al., 2020; Zhang et al., 2021), which can enforce the consistency of representations under different augmentations by contrasting positive pairs with negative ones. However, contrastive learning does not take the samples' semantic information and semantic similarity into account, where it simply treats two samples as a positive pair as long as they are at the same location of training examples that will be propagated through the network (i.e., in the same batch), and a negative pair when they are in different batches, without considering their semantic information. By doing this, the learned representation can be considerably affected. For example, simply adopting contrastive learning can push instances with the similar semantic meaning but in different batches far away from each other, and thus may break the semantic consistency between them.

To consider similarity between samples, clustering algorithms (Alwassel et al., 2020; Asano et al., 2020; Caron et al., 2020; Li et al., 2020d) can remedy the above problems, where clustering algorithms group similar instances (i.e., instances with the similar semantic meaning) into the same cluster, and push different instances away into different clusters. Therefore, contrastive learning and clustering learning complement each other, where adopting contrastive learning allows us to better explore mutual

information across modalities, and using clustering learning groups information with the similar semantic meaning together into the same cluster, which enables a consistent representation. More details are presented in Chapter 3.

After having a consistent representation for information from the same or different modalities with the similar semantic meaning, the other question arises: *how to achieve an effective interaction between this information?* More specifically, *how to effectively build connections between visual features from an image with corresponding semantic words from a sentence?* This is because a machine needs to understand the semantic meaning of a given text description and also identify specific image regions, and then it can generate an image with text-required objects and attributes, or modify specific image regions to match the given text description.

To achieve an effective interaction between information from different modalities, three approaches are proposed: (1) word-level attention-based connection, aided by a complementary word-level discriminator to provide fine-grained training feedback, (2) a text-image affine combination module, and (3) a memory-driven approach. These approaches enable the network to effectively correlate image regional visual attributes with corresponding semantic words. More details are presented in Chapter 4.

After having a framework to enable an effective interaction between multi-modal information, another question arises: *can this framework be easily deployed in most devices?* More specifically, *is it possible to develop a framework that does not rely on expensive computing resources?* This is mainly because when a network involves information from different modalities, the number of trainable parameters might need to considerably increase, which aims to allow the network to comprehensively learn to capture the correlations between multi-modal information with a domain gap. The requirements for having significant computing resources might greatly hinder the deployment of this framework, which is not practical for the implementation in real-world applications. To solve this problems, we propose to improve the capability of both generator and discriminator of GAN-based networks, and also rethink the architecture of discriminator in conditional GANs, instead of blindly increasing the feature dimensions of a network by having a large amount of trainable parameters. More details are presented in Chapter 5.

1.3 Outline

This thesis mainly focuses on solving aforementioned questions step by step: in Chapter 3, combining both contrastive learning and clustering learning together allows

to have a consistent representation for information with the similar semantic meaning, in Chapter 4, three different approaches are proposed to effectively correlate image visual attributes with corresponding semantic text descriptions, which allows a network to learn to understand the text and image information, and in Chapter 5, two solutions relieve the cost of computing resources, allowing our framework be easily implemented in various areas. Although the contributions of each chapter focuses on a question, respectively, but we actually can combine all these contributions together to have a lightweight network, which enables a consistent representation and also allows to achieve an effective interaction between pieces of information from different modalities.

To put into perspective the scope of these contributions, before diving into the details of this thesis, in Chapter 2, I provide the basic background knowledge on interaction between multi-modal information for deep neural networks, especially focusing on generative models.

At the end of each chapter, I present the conclusions that the findings in the chapter lead to. Finally, in Chapter 6, I provide general conclusions and perspectives on interaction between multi-modal information for deep neural networks, and also future directions that we are going to explore.

Chapter 2

Background on Multi-Modal Generative Methods

This thesis assumes that readers have basic knowledge of machine learning, especially deep neural networks. So, in this chapter, I provide the necessary background of multi-modal generative methods, including different network architecture, relevant components and techniques, and also the tasks that I used to evaluate the performance of the proposed approaches. Besides, I introduce the datasets and different evaluation metrics that are adopted in the experiments.

2.1 Types of Generative Model Architecture

We use deep generative models (Goodfellow et al., 2014; Mirza and Osindero, 2014; Kingma and Welling, 2013; Vincent et al., 2010) to evaluate the performance of our proposed approaches on achieving an effective interaction between multi-modal information. This is because deep generative models can transfer information from one domain to another, and according to the quality of output results, we can directly observe whether the proposed approaches can understand the semantic meaning of multi-modal information and whether they can effectively interact with each other, and thus evaluate the effectiveness of them.

2.1.1 Autoencoder

The general idea of *autoencoder* consists in setting an encoder and a decoder as neural networks, and learns the best encoding-decoding scheme using an iterative optimization process. So, at each iteration, we feed the autoencoder architecture (the encoder followed by the decoder) with some data, we compare the encoded-decoded

output with the initial data, and back propagate the error through the architecture to update the weights of the networks.

Therefore, intuitively, the overall autoencoder architecture (encoder + decoder) creates a bottleneck for the data, ensuring that only important information can pass through and be reconstructed. In the general framework shown in Fig. 2.1, the family e of considered encoders is defined by the encoder network, the family d of considered decoders is defined by the decoder network, and the search for encoders and decoders that minimize the reconstruction error is performed by gradient descent over the parameters of these networks.

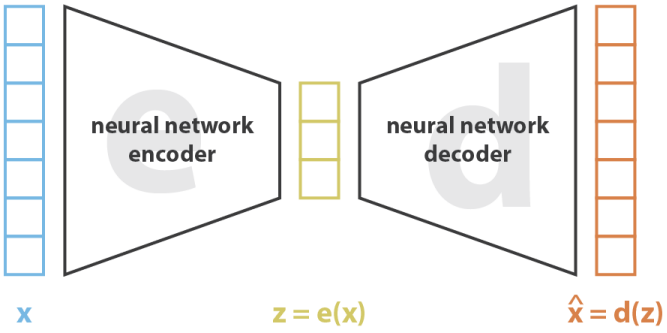


Figure 2.1: Architecture of the autoencoder.

The corresponding loss function is defined below, which aims to minimize the difference between given data x and the reconstructed data $\hat{x}$.

$$\text{Loss} = \|x - \hat{x}\|^2 = \|x - d(z))\|^2 = \|x - d(e(x))\|^2 . \tag{2.1}$$

2.1.2 Variational Autoencoder

To use the decoder of the autoencoder for generative purposes, it is crucial to ensure that the latent space is regular enough. A regular latent space ensures that the decoder can generate coherent and meaningful output from any point in the latent space. In other words, the decoder should be able to properly reconstruct the input data from all points in the latent space. Ensuring the regularity of the latent space is important to leverage the decoder for generative tasks, such as generating new data points that are similar to the input data.

One approach to achieve regularity in the latent space is to introduce an explicit regularization term during the training process. This is accomplished through the use of a *variational autoencoder* (VAE)(Kingma and Welling, 2013). The training of a VAE is regularized to prevent overfitting and ensure desirable properties in the latent

space, such as continuity, completeness, and smoothness. These properties enable smooth interpolation between data points, allow for the generation of any valid data point, and produce realistic data samples that exhibit smooth variations across the latent space, as illustrated in Fig. 2.2. The introduction of the regularization term in the VAE framework promotes the learning of a well-behaved latent space, which is beneficial for generative tasks.

Similar to a standard autoencoder, a variational autoencoder consists of an encoder and a decoder. It is trained to minimize the reconstruction error between the encoded-decoded data and the original data. However, in contrast to a standard autoencoder, a variational autoencoder introduces a regularization term in the latent space. This is achieved through a modification of the encoding-decoding process: rather than encoding an input as a single point, it is encoded as a distribution over the latent space. This modification allows for the modeling of the latent space distribution, enabling the generation of new data points by sampling from the learned distribution.

In practice, it is common to choose Gaussian normal distributions as the encoded distributions in VAEs. This choice allows the encoder to be trained to output the mean and covariance matrix of the Gaussian distribution, which facilitates the regularization of the latent space. By modeling the encoded distributions as Gaussian, the VAE framework can naturally incorporate the KL divergence regularization term and optimize the encoder to produce latent representations that follow a Gaussian distribution. This helps in achieving a more structured and regular latent space.

Based on this, the loss function minimized during the training of a VAE consists of two components: a reconstruction term, which measures the fidelity of the reconstructed output to the original input, and a regularization term, which encourages the latent space to exhibit desirable properties. The regularization term is typically expressed as the Kullback-Leibler (KL) divergence between the distribution returned by the encoder and a standard Gaussian distribution. It is worth noting that the KL divergence between two Gaussian distributions has a closed form expression that can be computed directly using the means and covariance matrices of the two distributions (Kingma and Welling, 2013). By incorporating this regularization term into the loss function, the VAE training process encourages the latent space to have a well-organized and smooth structure.

The corresponding loss function is defined below, which is used for both the encoder and the decoder to minimize the difference between given data x and the reconstructed data $\hat{x}$, including the KL divergence between the returned distribution N_{μ_x,σ_x} and a

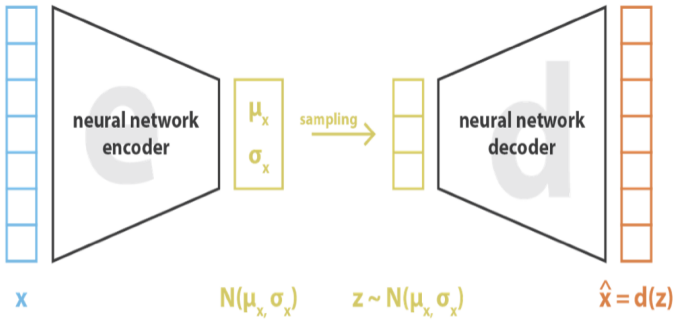


Figure 2.2: Architecture of the variational autoencoder.

standard Gaussian $N(0, 1)$.

$$\text{Loss} = \|x - \hat{x}\|^2 + \text{KL}\left[N_{\mu_x,\sigma_x}, N(0,1)\right] = \|x - d(z))\|^2 + \text{KL}\left[N_{\mu_x,\sigma_x}, N(0,1)\right]. \quad (2.2)$$

2.1.3 Generative Adversarial Networks

Generative adversarial networks (Goodfellow et al., 2014) consist of two models that are alternatively trained to compete in a two-player min-max game: the generator G is optimized to fool the discriminator D by reproducing the true data distribution p_{data}, and the discriminator D is optimized to distinguish real images from synthetic images generated by G. Concretely, this training procedure can be modeled by a min-max game over the value function $V(D, G)$:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))], \quad (2.3)$$

where x is a real image sampled from the real data distribution p_{data}, and z is a noise vector drawn from a distribution p_z (e.g., Gaussian or uniform distribution).

Goodfellow et al. (2014) proved in their paper that this min-max game could reach a global optimum if and only if $p_g = p_{\text{data}}$, where p_g is a probability distribution, implicitly defined from the generator G as the distribution of the samples $G(z)$ obtained when $z \sim p_z$. In practice, G is extremely poor at the beginning of training, and then D could reject samples with high confidence as these samples are clearly different from the training data. Therefore, they suggested that it would be better for the generator to maximize $\log(D(G(z)))$ rather than minimizing $\log(1 - D(G(z)))$.

2.1.4 Conditional Generative Adversarial Networks

Conditional GANs (Gauthier, 2015; Mirza and Osindero, 2014) is an extension of GANs, where both the generator and discriminator are provided with the additional

conditioning variable c, yielding $G(z, c)$ and $D(x, c)$. The c could be any kind of extra information, such as class labels or data from other modalities. This reformulates that images generated from G must not only look realistic, but also match its auxiliary information c.

In the generator, the auxiliary information c and the prior input noise $p(z)$ are combined together as a joint hidden representation. While, in the discriminator, the real sample x and c are concatenated together as inputs to a discriminative function. The objective of this two-player mini-max game would be:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x, c)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z, c)))]. \tag{2.4}$$

2.1.5 Multi-Stage Architecture

In a multi-stage architecture in GANs, the generator and discriminator are trained in a progressive manner, where they start with low resolution images and gradually increase the resolution until they reach the desired output size (Zhang et al., 2017a). This approach allows the network to learn the basic features of the image at a low resolution and then refine those features at higher resolutions, resulting in a more detailed and realistic final image.

The multi-stage architecture typically consists of several stages, where each stage corresponds to a specific resolution of the output image. At each stage, the generator and discriminator are trained separately on a dataset of images with the corresponding resolution. Once a stage is trained, the generator produces images of that resolution, which are used as input for the next stage.

The advantage of this approach is that it allows the network to learn more complex and realistic features at higher resolutions, while still maintaining the basic features learned at lower resolutions. This results in images that are not only more detailed but also more coherent and realistic.

2.2 Attention

An attention model in a neural network is a mechanism that allows the network to focus its processing resources on a subset of the input data at each time step, while suppressing other regions that are deemed less important. The attention can be used temporally in natural language processing, where the model attends to different parts of the input sequence at different time steps, or spatially in computer vision tasks, where the model attends to different parts of the input image or video frame.

The attention mechanism has been widely used in various research fields, including image captioning (Xu et al., 2015; Zhang et al., 2017b; Anderson et al., 2018; Salvador et al., 2016; Xu et al., 2015), machine translation (Bahdanau et al., 2014; Vaswani et al., 2017; Luong et al., 2015; Bahdanau et al., 2014; Wu et al., 2016), object detection (Oliva et al., 2003; Zhang et al., 2018b; Bello et al., 2019; Carion et al., 2020), visual question answering (Yang et al., 2016; Xu and Saenko, 2016; Nam et al., 2017; Anderson et al., 2018), and more. In image captioning, for example, the attention mechanism can be used to selectively attend to different regions of an image to generate a descriptive caption. Similarly, in machine translation, the attention mechanism can be used to focus on different parts of the input sentence when generating each word of the output translation.

The key advantage of the attention mechanism is its ability to capture task-relevant information and reduce the interference from less important information. By attending only to the relevant parts of the input, the model can effectively filter out noise and distractions, resulting in more accurate and efficient processing. Besides, the attention mechanism provides a way for the model to visualize and explain its decision-making process, which can be useful in applications where interpretability is important.

2.3 Contrastive Learning in GANs and Clustering Learning

Contrastive learning is a powerful technique for *self-supervised* representation learning (Oord et al., 2018; He et al., 2020; Chen et al., 2020; Zhang et al., 2021; Lee et al., 2021; Deng et al., 2020), which aims to learn representations that maximize the similarity between positive pairs of samples while minimizing the similarity between negative pairs. In other words, the objective of contrastive learning is to bring the representations of the same data point closer together in the latent space, while pushing apart the representations of different data points. This is achieved by mapping each data point to a low-dimensional latent space, where the distance between positive pairs (e.g., two different augmentations of the same image) is minimized, while the distance between negative pairs (e.g., two images from different classes) is maximized.

By training a model to discriminate between positive and negative pairs in this way, the resulting representations are expected to capture meaningful features that are invariant to variations in the input data, such as different viewpoints or lighting conditions. Contrastive learning has been shown to be effective in a variety of applications, including image and speech recognition, language modeling, and reinforcement learning.

XMC-GAN (Zhang et al., 2021) used intra-modality and inter-modality contrastive learning in text-to-image synthesis. ContraGAN (Kang and Park, 2020) adopted contrastive learning in class-conditional image generation. DiscoFaceGAN (Deng et al., 2020) incorporated contrastive learning in face generation to enable a better disentanglement. CUT (Park et al., 2020) used positive pairs from the same image patches in both input and output images for image-to-image translation.

Differently, we further extend contrastive learning with clustering learning in conditional GANs to consider samples' semantic information and similarity between different samples. Clustering learning can discriminate between groups of similar instances during training, which is to group together similar samples based on some similarity metric or distance measure, while separating dissimilar samples into different clusters. In other words, the goal of clustering learning is to partition a set of unlabeled data points into distinct groups, where each group represents a subset of points that share similar characteristics or belong to the same underlying class or cluster. This is typically done by defining a similarity or dissimilarity metric between data points, and using this metric to iteratively group similar points together into clusters.

Clustering learning is commonly used in unsupervised machine learning, where the goal is to find structure or patterns in data without any prior knowledge of the underlying labels or categories. It has been applied in a variety of domains, including image and text analysis, social network analysis, and customer segmentation. Caron et al. (2018) and Yan et al. (2020) kept the clustering step independent of the representation learning phrase, while Asano et al. (2020, 2019), and Van Gansbeke et al. (2020) jointly learned visual representations and cluster assignments. As for implementing clustering in multi-modal tasks, XDC (Alwassel et al., 2020), SeLaVi (Asano et al., 2020), and MCN (Chen et al., 2021) focused on pre-trained feature extractors for audio and video, and aimed to produce effective multi-modal representations used for different downstream tasks.

2.4 Memory Bank

The memory bank is a storage that stores pre-processed information, such as image features and semantic data, which can be utilized by a neural network to gain specific advantages, such as faster computation, improved performance, and better generalization. By accessing pre-computed information, the network can save computation time, access additional data, and learn from a more diverse set of examples.

Researchers have applied the memory bank to various tasks, such as image generation, appearance determination, image patch retrieval, and conditional image extrapolation. For instance, Qi et al. (2018) proposed a memory-driven approach to image synthesis from text, using a memory bank to store pre-processed information such as image features and additional semantic data. TTseng et al. (2020) introduced RetrieveGAN, a memory-driven approach that utilizes a memory bank of pre-computed image patches for differentiable retrieval during image synthesis. Furthermore, Li et al. (2021) proposed a segmented StyleGAN that leverages a memory bank of stored image crops to synthesize new images guided by structured text inputs, achieving controllable image synthesis. Additionally, Liang et al. (2020) proposed CPGAN, a memory-driven approach that utilizes a memory structure to parse textual content and generate images while preserving their content.

By integrating the memory bank, a memory-driven approach can be constructed that combines the advantages of both parametric and non-parametric techniques. The non-parametric aspect is a memory bank of pre-processed information generated from the training data, while the parametric component is a neural network. This approach can enable the end-to-end training of powerful models while also allowing the network to use vast datasets of information during inference, leading to faster and more accurate predictions.

2.5 Types of Tasks

To comprehensively evaluate the performance of our proposed approaches, we mainly focused on three tasks, including *text-to-image generation*, *story visualization*, and *text-guided image manipulation*, where each task is able to evaluate the performance of proposed approaches on specific questions described in Section 1.2

2.5.1 Text-to-Image Generation

Text-to-image generation is a task in artificial intelligence and computer vision that involves generating realistic images from textual descriptions. This task requires a machine learning model to learn the mapping between natural language descriptions and corresponding visual content. It involves understanding the semantics of the language and synthesizing an image that accurately represents the described scene. Text-to-image generation has applications in various domains such as virtual reality, gaming, and e-commerce, where automatic generation of high-quality and diverse images is desired.

Deep learning techniques such as pixelCNN (Van den Oord et al., 2016; Gregor et al., 2015), variational autoencoders (VAEs) (Kingma and Welling, 2013), and generative adversarial networks (GANs) (Goodfellow et al., 2014) have been widely used in text-to-image generation. The field has witnessed significant progress in recent years with the introduction of various methods. For instance, AlignDRAW (Mansimov et al., 2015) utilized an attention mechanism to draw image patches in multiple stages based on caption words. Approximate Langevin methods were proposed by Nguyen et al. (2017) for synthesizing images from text. GAN-INT-CLS (Reed et al., 2016b) was the first GAN-based method developed for this task. StackGAN (Zhang et al., 2017a) introduced a stacked architecture to generate realistic images in a coarse-to-fine manner. AttnGAN (Xu et al., 2018) and ControlGAN proposed in Section 4.1 employed word-level attention to capture text-image relationships. DMGAN (Zhu et al., 2019) utilized a memory module to refine image contents. MirrorGAN (Qiao et al., 2019a) proposed a text-image-text feedback loop to improve text-image alignment. ObjGAN (Li et al., 2019b) and OPGAN (Hinz et al., 2019) leveraged additional object-level annotations such as bounding boxes or masks. Johnson et al. (2018) and Ashual and Wolf (2019) incorporated scene graphs to provide logical relations between objects. XMC-GAN (Zhang et al., 2021) introduced a contrastive learning-based text-to-image generation method. DF-GAN (Tao et al., 2020) proposed a deep fusion module to combine text and image features. DALL-E (Ramesh et al., 2021) achieved zero-shot text-to-image generation based on an autoregressive transformer. GLIDE (Nichol et al., 2021) utilized a diffusion model for text-to-image generation and text-guided image editing. These methods have contributed to the advancements in text-to-image generation, each with its unique approach and capabilities.

State-of-the-art methods. Several recent studies have proposed state-of-the-art approaches for high-resolution image synthesis and text-to-image synthesis. Rombach et al. (2022) presented a novel method that utilizes latent diffusion models to generate high-resolution images progressively through a series of noise-perturbed steps. Gu et al. (2022) introduced a technique that combines a diffusion process with vector quantization to produce high-quality images from textual descriptions. Sauer et al. (2023) proposed a new architecture that integrates GANs with a text encoder, enabling efficient and large-scale text-to-image synthesis with high-quality results. These methods represent the current state-of-the-art in their respective domains.

Conclusion. Text-to-image generation is a task that involves generating realistic images from textual descriptions while maintaining semantic alignment. This task serves as a suitable evaluation for the network's ability to understand the semantic

meaning of a given text, as it requires the synthetic output image to accurately represent the desired visual attributes and objects described in the text.

2.5.2 Story Visualization

Story visualization aims to generate a series of images from a given multi-sentence story, and different output images within a story should be consistent with each other. StoryGAN (Li et al., 2019c) first introduced this task and proposed a sequential conditional generative adversarial network. Built on top of StoryGAN, CP-CSV (Song et al., 2020) added another segmentation mask pipeline to keep character consistency, and DUCO (Maharana et al., 2021) and VLC (Maharana and Bansal, 2021) utilized video captioning networks to improve text-image alignment. As all these methods have a similar architecture, which contains two sets of generative adversarial networks, one for image quality, and one for story consistency, and thus computational requirements to hold and train their networks are large.

State-of-the-art methods. Maharana et al. (2022) proposed a new model called StoryDALL-E that uses a DALL-E-like transformer-based architecture to generate images that depict a given story, and achieved state-of-the-art performance on the TextCaps benchmark dataset. Rahman et al. (2022) proposed a model that generates consistent and diverse story visualizations by conditioning on a visual memory, which can be an image or a set of images, and incorporating memory-based attention and a multi-level story structure. Pan et al. (2022) proposed an auto-regressive latent diffusion model for synthesizing coherent and diverse stories from text prompts, which can capture long-range dependencies and generate high-quality samples, and outperforms previous state-of-the-art methods on various evaluation metrics.

Overall, each method contributes unique approaches to generating coherent and diverse story visualizations, but the auto-regressive latent diffusion model currently performs the best due to its ability to capture long-range dependencies and generate high-quality samples.

Conclusion. The task of story visualization involves generating a series of synthetic images that are semantically aligned with a given series of sentences. Unlike text-to-image generation, where a single image is generated from a single textual description, story visualization requires generating multiple images that maintain global consistency throughout the story. This task is more challenging and computationally intensive compared to text-to-image generation since it involves generating multiple images.

Story visualization serves as an appropriate task for evaluating the quality of representations for both text and image information. The ability to produce coherent

and consistent visualizations that align with the textual content demonstrates the quality of the learned representations. If the representations are not consistent for both modalities, the output results of story visualization may be significantly affected.

Additionally, story visualization can be used to evaluate the feasibility of building lightweight networks with a small number of trainable parameters while still achieving competitive performance compared to larger networks. The task provides a suitable benchmark to assess whether a compact network can effectively capture the complex relationships between text and image modalities.

In summary, story visualization is a challenging task that requires generating a series of semantically aligned images from a series of sentences. It serves as a valuable evaluation metric for assessing the quality of representations and the performance of networks in capturing the interplay between text and image information.

2.5.3 Text-Guided Image Manipulation

Text-guided image manipulation aims to modify a real image while keeping text-irrelevant contents unchanged, using a given natural language description. At the time of writing this paper, only a few studies have worked on this task. Brock et al. (2016) introduced a hybridization model using VAE (Kingma et al., 2014) and GANs (Goodfellow et al., 2014), which allowed to achieve an accurate reconstruction without loss of feature quality. Zhu et al. (2016) proposed to change the color and shape of an object by manipulating latent vectors. Dong et al. (2017) proposed a GAN-based encoder-decoder architecture to disentangle the semantics of both input images and text descriptions. Nam et al. (2018) built a model that focused on the multi-modal learning on both text descriptions and input images, and also proposed a text-adaptive discriminator to explore word-level text-image matching scores, to provide the generator with fine-grained training signals.

State-of-the-art methods. Recently, there have been significant advancements in diffusion-model-based text-guided image manipulation, which can generate realistic modified images using text. Kim and Ye (2021) proposed a text-guided image manipulation method based on diffusion models that incorporates perceptual metrics to better preserve the content and style of the input image. The proposed method outperforms existing methods on several benchmark datasets. Couairon et al. (2022) proposed Diffedit, a diffusion-based method that uses mask guidance to enable semantic image editing. Kawar et al. (2022) utilized a novel hierarchical diffusion process and leveraged the power of pre-trained GAN models, achieving semantic and structural consistency by incorporating a mask-guided attention mechanism and a semantic loss.

Different classes of methods have been proposed for text-guided image manipulation, with some focusing on accurate reconstruction and manipulation of input images, while others aim to disentangle the semantics of input images and text descriptions. Recent works on diffusion-model-based text-guided image manipulation have shown significant advancements in generating realistic modified images using text. These methods leverage diffusion models and mask guidance, as well as perceptual metrics and pre-trained GAN models, to achieve semantic and structural consistency. Overall, diffusion-model-based methods seem to perform better due to their ability to preserve image content and style while allowing for semantic manipulation.

Conclusion. As text-guided image manipulation involves both image and text inputs, it is a suitable task for evaluating the network's ability to effectively correlate and fuse information from different domains. If the network fails to do so, the modified image may lack the desired image attributes specified in the text, and irrelevant image contents may also be modified unnecessarily.

2.6 Types of Datasets

In this section, we show the details of different datasets that are used to evaluate the effectiveness of our proposed approaches.

2.6.1 CUB Bird

CUB (Caltech-UCSD Birds-200-2011) (Wah et al., 2011) is a publicly available dataset of bird images intended for fine-grained visual categorization tasks. It includes 11,788 annotated images of birds, belonging to 200, mostly North American, bird species. Each image is annotated with a rough segmentation, a bounding box, and binary attribute annotations, where binary attribute annotations refer to a set of binary labels assigned to each image that indicate the presence or absence of certain attributes in the image. For example, an image may have binary attribute labels indicating whether the bird in the image has a "black crown," "white belly," "long tail," "pointed wings," etc. In the dataset, it contains 8,855 training images and 2,933 test images, and each image has 10 corresponding text descriptions.

2.6.2 MS COCO

MS COCO (Microsoft Common Objects in Context) (Lin et al., 2014) is a large-scale object detection, segmentation, and captioning dataset, which contains annotations

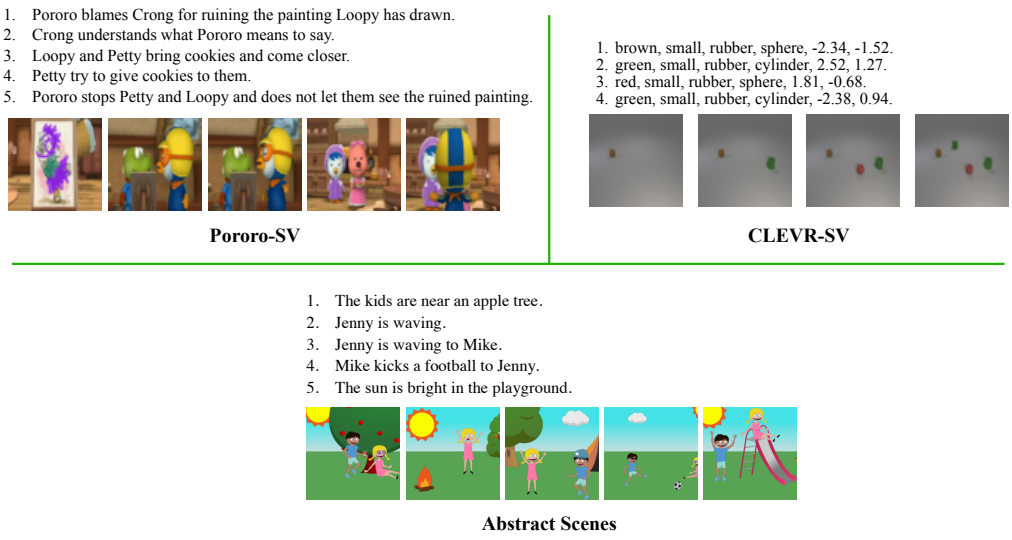


Figure 2.3: Illustration of Pororo-SV, CLEVR-SV, and Abstract Scenes.

that you can use to train machine learning models to recognize, label, and describe objects. It contains 82,783 training images and 40,504 validation images, and each image has 5 corresponding text descriptions.

2.6.3 Pororo-SV

Pororo-SV is a modified version of the PororoQA dataset, which was originally created for video question answering tasks (Kim et al., 2017). For Pororo-SV, a single image frame was selected from a video clip, and a story was formed by combining five consecutive images with their corresponding text descriptions. In total, 15,336 story-description pairs were generated, with 13,000 pairs used for training and 2,336 pairs reserved for the test set.

2.6.4 Abstract Scenes

Abstract Scenes (Zitnick and Parikh, 2013; Zitnick et al., 2013) was proposed for studying semantic information, which contains over $1,000$ sets of 10 semantically similar scenes of children playing outside. The scenes are composed with 58 clip-art objects, and there are six sentences describing different aspects of a scene.

2.6.5 CLEVR-SV

CLEVR dataset (Johnson et al., 2017) was initially introduced for the task of visual question answering. CLEVR-SV dataset is a variation of CLEVR dataset, where

image sequences were generated by adding objects one by one, starting from randomly assigned layouts. The dataset comprises 10,000 image sequences for the training set and 3,000 for the test set.

2.7 Evaluation Metrics

In this section, we give details about different evaluation metrics used in the thesis to quantitatively evaluate the effectiveness of different proposed approaches.

2.7.1 Inception Score

Inception score (IS) (Salimans et al., 2016) is a metric used to evaluate the quality of generated images produced by generative models, particularly generative adversarial networks (GANs). It measures the diversity and quality of the generated images by examining how well the generated images match the distribution of images in the training set. The IS is calculated by feeding the generated images to a pre-trained Inception network, which is a convolutional neural network (CNN) used for image classification. The output of the Inception network is then used to calculate the entropy of the predicted label distribution for each generated image. The IS is computed by taking the exponential of the mean of the Kullback-Leibler (KL) divergence between the marginal label distribution of the generated images and the same distribution for the training set. A higher IS value indicates that the generated images are diverse and of high quality, and are similar to the training set.

However, IS has some drawbacks: (1) Hinz et al. (2019) showed that IS is less consistent with human evaluation; (2) IS has a limitation in that it does not assess the similarity between objects of the same class, which can result in a network achieving a high IS despite exhibiting an intra-class mode dropping behavior; and (3) Li et al. (2019b) showed that IS can easily overfit within the context of text-to-image generation and can be exaggerated by simply increasing the batch size at the end of training.

Implementation. The Inception score is a metric commonly used in text-to-image generation and text-guided image manipulation tasks to evaluate the diversity and quality of the generated images produced by generative models.

2.7.2 Fréchet Inception Distance

Fréchet Inception Distance (FID) (Heusel et al., 2017) is a metric for evaluating the similarity between two sets of images. It was proposed as an improvement over

the Inception score, which only measures the quality and diversity of generated images but does not take into account their similarity to real images. FID uses the activations of a pre-trained Inception model to calculate the distance between the feature representations of the real and generated image sets. The lower the FID score, the closer the generated images are to the real ones in terms of their visual appearance and structure. FID has been widely used as a reliable metric for evaluating the quality of generative models, particularly in the context of generative adversarial networks.

Implementation. The Fréchet Inception Distance is utilized in text-to-image generation, text-guided image manipulation, and story visualization to evaluate the quality of generative models.

2.7.3 Fréchet Story Distance

Fréchet story distance (FSD) was proposed for story visualization by Song et al. (2020). It is based on the principle of Fréchet Inception Distance (FID) and utilizes the R(2+1) backbone model proposed by Tran et al. (2018). FSD evaluates the similarity between two sequences of images and incorporates the temporal consistency by leveraging the spatial and temporal convolutions of the R(2+1) architecture. The R(2+1) architecture is a convolutional neural network design that combines both spatial and temporal convolutions to effectively capture appearance and motion information in videos.

Implementation. The Fréchet story distance is used in story visualization to evaluate the image quality of a sequence of consistent images.

2.7.4 R-Precision

To measure the semantic alignment between the synthetic image and the corresponding given text description, R-precision (Xu et al., 2018) is adopted in the thesis. R-precision is calculated by retrieving relevant text descriptions given an image query. To measure the relevance between the text and the image, cosine similarity between text and image features is adopted. Thus, we compute a global image vector and 100 candidate sentence vectors, where the 100 candidate sentence vectors contain R number of ground-truth text descriptions that correctly describe the image, and $100 - R$ randomly chosen mismatched descriptions. For each image query, if n results in the top R ranked retrieval text descriptions are relevant, then the R-precision is n/R. In the paper, we measure the top-1 R-precision (i.e., $R = 1$).

Implementation. R-precision is used in text-to-image generation to evaluate the semantic alignment between the synthetic image and the given sentence.

Chapter 3

Consistent Representations for Multi-Modal Information

In this chapter, I mainly focus on building consistent representations for information from different modalities with similar semantic meaning, e.g., text and image, which allows the generative module to effectively correlate image visual attributes with corresponding semantic words from a given sentence, and then achieves an effective interaction between them. To achieve this, we propose to utilize both contrastive learning and clustering learning, and adopt story visualization task to evaluate our proposed approaches. In the following, we first show the background of story visualization, and then introduce our proposed contrastive- and clustering-based approach to build consistent representations for multi-modal information with the similar semantic meaning.

3.1 Background of Story Visualization

There has been a great progress in text-to-image generation (Zhang et al., 2018a; Xu et al., 2018; Qiao et al., 2019b; Hinz et al., 2019; Zhu et al., 2019; Li et al., 2022b), due to realistic image generation using generative adversarial networks (GANs) (Goodfellow et al., 2014). The task of story visualization (Li et al., 2019c; Song et al., 2020; Maharana et al., 2021) is a more challenging variation of it, as we need to generate a sequence of story images given a multi-sentence story, and further require output images to be consistent, e.g., having a similar background or objects' appearances in all images within the same story.

As story visualization requires synthetic story images to be realistic and also consistent, current methods (Li et al., 2019c; Song et al., 2020; Maharana et al., 2021; Maharana and Bansal, 2021) implement two GANs, one for image quality, and one for

story consistency. Also, to ensure the quality of results, Song et al. (2020) required segmentation masks, and Maharana et al. (2021) and Maharana and Bansal (2021) adopted additional auxiliary captioning networks. However, all these works do not consider the semantic similarity between instances (i.e., training samples), including paired instances provided by the dataset, e.g., text-image pairs, and unpaired ones but having the similar semantic meaning. So, these methods may fail to make full use of mutual information between paired instances and also unpaired ones but having the similar semantic meaning, which may push similar instances far away from each other, and thus cause distance between them in the joint space. For example, in a dataset containing text-image pairs of different animals, a method that only considers the visual similarity between images and text may fail to recognize that two different images of cats or two different descriptions of cats are semantically similar. This could lead to pushing apart instances of similar animals in the joint space, and ultimately to a decrease in the overall performance of the model.

Based on this, we propose to adopt both contrastive and clustering learning in the generative network, where contrastive learning can maximize the mutual information between paired instances provided by the given dataset, and clustering learning can group instances with similar semantic meaning into the same cluster, and push dissimilar ones far away from each other. By doing this, a structured joint space can be built, where instances with similar semantic meaning can be closely grouped together to ensure a consistent representation, regardless of their modalities.

3.2 Contrastive- and Clustering-Based Approach for the Consistent Representation Construction

In this section, we give more details about our proposed contrastive- and clustering-based approach in the consistent representation construction.

3.2.1 Model Overview

The complete architecture is shown in Fig. 3.1. Differently from current methods (Li et al., 2019c; Song et al., 2020; Maharana et al., 2021; Maharana and Bansal, 2021), which have two generators and two discriminators, with the help of segmentation masks (Song et al., 2020) or auxiliary captioning networks (Maharana et al., 2021; Maharana and Bansal, 2021), the network of our approach is made up with a generator and a discriminator. The generator takes text embeddings encoded by a text encoder from a given multi-sentence story, and a noise vector sampled from a Gaussian

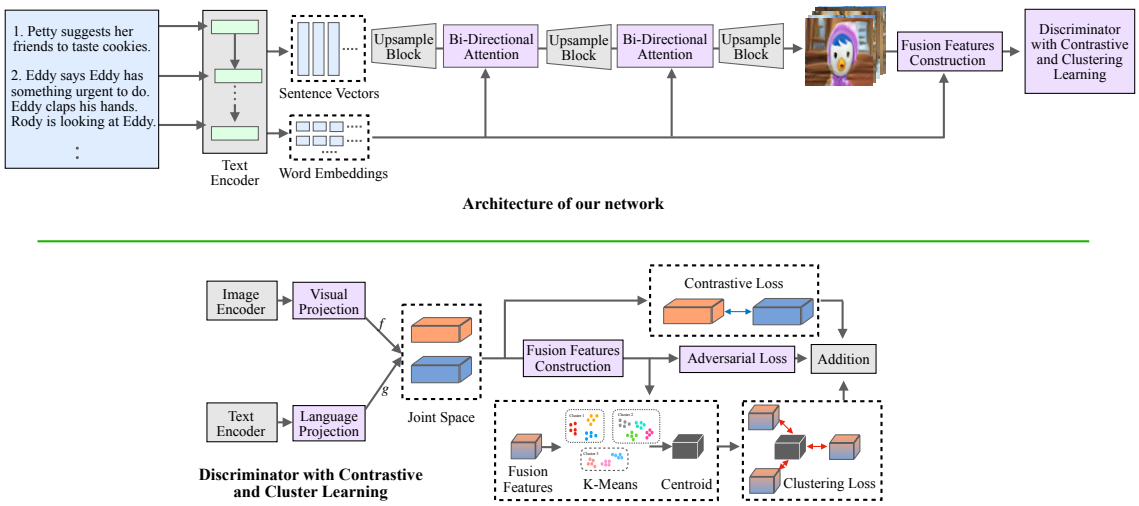


Figure 3.1: Illustration of the proposed framework.

distribution as inputs. First, the noise vector and text embeddings are fed into a fully connected network with a reshape operation to produce initial story image features at 4×4. Then, we apply a series of upsampling blocks to upsample image features into the desired resolution, generating all story images at the same time. The discriminator consists of a series of downsampling blocks, and evaluates the quality of given fusion features, providing training feedback to the generator. Meanwhile, our discriminator also implements contrastive learning to maximize mutual information between language and visual information, and adopts clustering learning to capture semantic similarity between instances.

3.2.2 Contrastive Learning for Story Visualization

Contrastive learning is a powerful scheme for self-supervised representation learning (Oord et al., 2018; He et al., 2020; Chen et al., 2020; Zhang et al., 2021). It enforces the consistency of image representations under different augmentations by contrasting positive pairs with negative ones.

Here, we consider four types of correspondences: (1) between story sentences and story images, (2) between single sentences and single images, (3) between words and image regions, and (4) between words and story image regions. The goal in each case is to maximize the mutual information between the corresponding pairs. We use global representations to capture the semantics of the text and images. Specifically, a sentence is represented by a global vector that combines the word embeddings within the sentence, while an image is represented by a global vector. Similarly, a story

sentence is represented by a global vector that combines the sentence vectors within the story, and a story image is represented by a global vector that combines the image vectors within the story. By maximizing the mutual information, we aim to establish meaningful and informative correspondences between the different elements of the text and image data.

For simplicity, in the following, V represents the visual input (i.e., story image, image, or image regions), v represents corresponding visual features, extracted by an image encoder, S represents the language input (i.e., story sentence, sentence, or words), and s represents corresponding text features, encoded by a text encoder. In this task, the image encoder takes an image as input to extract image regional features and a global image feature, and then we average a series of global image features within the same story as a story image feature. The text encoder takes multiple sentences from a story as input to encode a global story sentence vector, sentence vectors, and word embeddings.

In our work, we propose to use a masked margin softmax function (MMS) (Ilharco et al., 2019; Chen et al., 2021) to define the similarity between representations from visual and language information, with respect to their learned embeddings' dot product within a batch N. This is different from XMC-GAN (Zhang et al., 2021), which adopts the normalized temperature-scaled cross entropy loss (NT-Xent) (Chen et al., 2020). Using MMS instead of NT-Xent is because NT-Xent may miss opportunities to learn against a wider set of negative examples, namely, all those in the batch that are not known to be positively associated, while MMS can exploit these additional negatives.

Given a set of pairs of associated story image and story sentence, we first construct parameterized mappings that derive embedding representations from extracted visual and language features. We use the transform $f : v \rightarrow \mathbb{R}^D$ to convert given visual features into a joint space $\mathbb{R}^D$, and the transform $g : s \rightarrow \mathbb{R}^D$ to map given language features into the same space. Thus, for a pair of information (e.g., single sentence and single image), the loss $\mathcal{L}_{\text{cont}}$ is:

$$
\begin{aligned}
-\frac{1}{N}\sum_{i=1}^{N}\Bigg[&\left(\log\frac{\exp(f(v_i)\cdot g(s_i)-\delta)}{\exp(f(v_i)\cdot g(s_i)-\delta)+\sum_{j=1,j\neq i}^{N}\exp(f(v_j)\cdot g(s_i))}\right)\\
&+\left(\log\frac{\exp(f(v_i)\cdot g(s_i)-\delta)}{\exp(f(v_i)\cdot g(s_i)-\delta)+\sum_{k=1,k\neq i}^{N}\exp(f(v_i)\cdot g(s_k))}\right)\Bigg],
\end{aligned}
\tag{3.1}
$$

where N is the number of training samples, and δ is a hyperparameter. Eq. 3.1 consists of two fractions: the numerator, which aims to maximize the similarity between paired vectors v_i and s_i, and the denominator, which aims to minimize the similarity between

a set of negative examples. By projecting all features to the same space and maximizing their pairwise similarities, we ensure that the mutual information is maximized between corresponding pairs. This helps in generating output results that are both realistic and consistent, as the learned representations capture meaningful relationships between the text and image data.

3.2.3 Clustering Learning for Story Visualization

However, contrastive learning does not take into account the semantic information and similarity of unpaired samples. It simply treats two samples as a positive pair as long as they are in the same location of the training samples (i.e., in the same batch), and a negative pair when they are in different locations (i.e., in a different batch), without considering their semantic information. This can considerably affect the learned representation. For example, instances within the same story have the similar semantic information, while simply adopting contrastive learning can push these instances far away from each other, and thus break the semantic consistency between them. Let us consider two sentences from the same story: Sentence 1: "John went to the store to buy some groceries." and Sentence 2: "He picked out a carton of eggs and a loaf of bread." These sentences share a common context and are semantically related, as they both describe John's trip to the store to buy groceries. However, if we only use contrastive learning without considering their shared context, the embedding of Sentence 1 could end up far away from the embedding of Sentence 2, making it more difficult for the model to understand the overall story.

By building connections between different-modal pieces of information (in this case, the sentences within each story), we can help ensure that semantically related instances remain close together in the embedding space, preserving the semantic consistency and improving the overall quality of the learned representations.

To consider similarity between samples, clustering algorithms (Alwassel et al., 2020; Asano et al., 2020; Caron et al., 2020; Li et al., 2020d) can remedy the above problems, where clustering algorithms group similar instances (i.e., instances with similar semantic meaning) into the same cluster in the shared space $\mathbb{R}^D$, and push different instances away into different clusters.

One problem arising from clustering learning is that story visualization involves multi-modal features (i.e., language and visual features), if we simply cluster these features into different clusters, these multi-modal features may be only grouped into their own modalities separately. That is, visual instances are only clustered with visual instances, and language instances with language instances Chen et al. (2021).

So, it is important to have an approach that can pull multi-modal instances together in a joint space. To address this, we suggest to feed merged features into clustering algorithms, which are created by fine-grained word embeddings and image regional features. In the following, we first explore fusion features, and then introduce both online and offline clustering learning using fusion features in story visualization.

3.2.3.1 Fusion Features

First, we define the inputs for the clustering algorithm. If we simply cluster multi-modal features into different clusters, these features may be only grouped into their own modalities separately Chen et al. (2021). To address this, we use fusion features $X \in \mathbb{R}^{(H*W)\times L}$ as the input for clustering learning:

$$X = f(v_{\text{region}}) \cdot g(s_{\text{word}})^T, \tag{3.2}$$

where T denotes the transpose, v_{region} represents image regional features, s_{word} represents word embeddings, $H * W$ denotes the total number of spatial locations in image regional features, and L denotes the total number of words in a sentence. Here, we take the matrix multiplication over visual and language fine-grained features to represent a multi-modal instance, which preserves visual and language information to ensure a more precise clustering.

3.2.3.2 Online K-Means Clustering

We adopt the standard clustering algorithm K-means in the training process. To implement it, we cluster fusion features into k distinct groups. So, the objectives of the clustering algorithm are to partition N fusion features X_1, X_2,..., X_N into k ($< N$) sets $S = \{S_1, S_2, \ldots, S_k\}$, and to minimize the within-cluster distances:

$$\arg\min_{S} \sum\nolimits_{i=1}^{k} \sum\nolimits_{X \in S_i} \|X - C_i\|^2, \tag{3.3}$$

where C_i is the mean of fusion feature points in S_i. To have enough instances and to cover variant semantic information for clustering, following Chen et al. (2021), we use fusion features from the previous training samples to gather sufficient instances for online clustering learning.

So, we get k centroids $C_1, C_2, \ldots, C_k$, and then we use these centroids as contrastive loss reference targets. This target pulls the fusion features belonging to the same cluster closer to the centroid, and pushes the features far away from the other centroids:

$$\mathcal{L}_{\text{cluster}} = -\frac{1}{N} \sum\nolimits_{i=1}^{N} \log \frac{\exp(\cos(X_i, C_X)/\tau)}{\sum_{j=1}^{k} \exp(\cos(X_i, C_j)/\tau)}, \tag{3.4}$$

where N is the number of training samples, and C_X is the centroid of X_i. Finally, the network learns to cluster similar fusion features created by both textual and visual inputs to be closer to their centroids, and thus learns to capture semantic similarity across modalities.

3.2.3.3 Offline K-Means Clustering

In online K-means clustering, the centroids are not fixed and are adjusted during the training of the GAN, which may take time to find the centroids. Instead, we precompute the centroids and fix them during training. To do this, we first pretrain an image encoder (e.g., VGG-16 (Simonyan and Zisserman, 2014)) and a text encoder (e.g., LSTM) to directly map both visual and textual inputs into a joint semantic space $\mathbb{R}^D$. The objectives to train both the image and text encoders are defined below:

$$\begin{aligned}\mathcal{L}_{\text{pretrain}} =& \mathcal{L}_{\text{cont}}(v^r_{\text{s\_image}}, s_{\text{s\_sent}}) + \mathcal{L}_{\text{cont}}(v^r_{\text{image}}, s_{\text{sent}}) \\ &+ \mathcal{L}_{\text{cont}}(v^r_{\text{regions}}, s_{\text{words}}) + \mathcal{L}_{\text{cluster}}(X^r) \\ &+ \mathcal{L}_{\text{cont}}(v^r_{\text{s\_regions}}, s_{\text{words}}),\end{aligned} \tag{3.5}$$

where r represents features extracted from a real sample, $\text{cont}(v, s)$ denotes applying contrastive losses between visual representation v and language representation s, $\mathcal{L}_{\text{cluster}}(X)$ represents applying clustering learning on fusion features X, which are created by word embeddings and image regional features, $v_{\text{s\_image}}$ denotes visual features from a story image, $s_{\text{s\_sent}}$ denotes text features from a story sentence, and $v_{\text{s\_regions}}$ denotes regional features for a story image. To pretrain both text and image encoders, we use paired story images and sentences from the given dataset. Therefore, after finishing the pretraining of both text and image encoders, we can find the centroids and then fix them during the training of the generative model to speed up the training process.

3.2.4 Another Components

In addition to our proposed contrastive- and clustering-based approach to ensure consistent representation construction, we also suggest feeding merged features into the discriminator. The discriminator then compares the quality of the fusion features created by word embeddings and image regional features extracted from synthetic images to the fusion features created by the word embeddings and image regional features extracted from ground-truth images. We believe that feeding fusion features into the discriminator can help regularize training, stabilize clustering learning with fusion features as inputs, and improve the overall performance of the model.

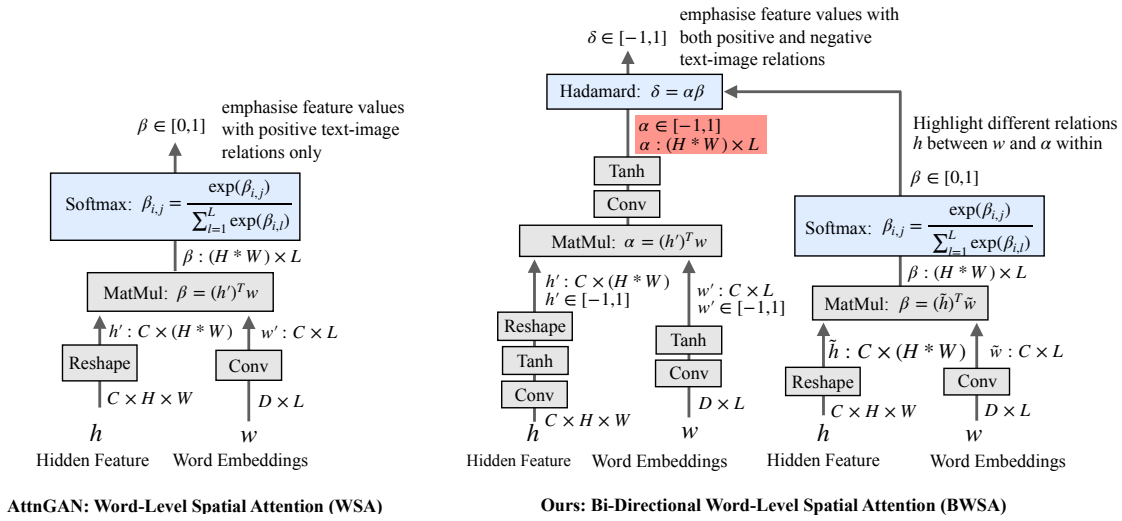


Figure 3.2: Comparison between WSA and BWSA.

As the discriminator uses word-level information to provide fine-grained training feedback, we propose to use it in the generator to improve its ability to capture relations between language and visual features. To achieve this, we adopt word-level spatial attention (WSA) and further extend it to comprehensively capture both positive and negative relations between image regions and different words. We call this extension *bi-directional word-level spatial attention* (BWSA), as shown in Fig. 3.2. Positive effects of BWSA work similarly to WSA to highlight image regions corresponding to semantic words, while negative effects work as a complementary component to capture relations that cannot be observed by the positive part. WSA implements the softmax function to produce a weight matrix, and thus almost all values in WSA are greater than zero, meaning that the attention utilizes the scales of positive values to highlight or ignore different image regions. However, not all words have a positive impact on image regions, even if the impact is small. Therefore, to keep both negative and positive effects, BWSA only highlights or ignores the relations between words and image regions, rather than the actual values of hidden features.

3.2.4.1 Discriminator

The discriminator consists of a series of downsampling blocks, and inputs for our discriminator are the fusion features (see Eq. 3.2). Fusion features are created by fine-grained word embeddings and image regional features, extracted from a real sample drawn from the dataset. So, the complete objectives to train the discriminator

$\mathcal{L}_D$ are defined as follows:

$$\begin{aligned}\mathcal{L}_D =& \mathcal{L}_{\text{cont}}(v^r_{\text{s\_image}}, s_{\text{s\_sent}}) + \mathcal{L}_{\text{cont}}(v^r_{\text{image}}, s_{\text{sent}}) \\ &+ \mathcal{L}_{\text{cont}}(v^r_{\text{regions}}, s_{\text{words}}) + \mathcal{L}_{\text{cont}}(v^r_{\text{s\_regions}}, s_{\text{words}}) \\ &+ \mathcal{L}_{\text{cluster}}(X^r) + \mathcal{L}^D_{\text{GAN}},\end{aligned} \tag{3.6}$$

$$\mathcal{L}^D_{\text{GAN}} = -\frac{1}{L}\sum\nolimits_{i=1}^{L} \log\left(\mathbb{E}\left[\log(D(X^r_i))\right] + \mathbb{E}\left[\log(1 - D(X^f_i))\right]\right), \tag{3.7}$$

where r represents features extracted from a real sample, L is the length of a story (i.e., the number of text-image pairs in a story), X^r are real fusion features created by word embeddings and image regional features extracted from the real image, X^f are fake fusion features created by word embeddings and image regional features extracted from the fake image. Here, following Zhang et al. (2021), we use the real stories and their descriptions to train discriminator projection heads f and g (see Section 3.2.2). This is because the generated images may have a poor quality, especially at the start of training, and using such low-quality synthetic image and sentence pairs can hurt the training of the projection heads f and g. So, the contrastive losses from fake images are only applied to the generator.

3.2.4.2 Generator

Our generator only consists of a series of upsampling blocks to generate story images at the desired resolution from given descriptions. Meanwhile, it uses BWSA to utilize word-level information to improve its capabilities of capturing different relations. The objectives to train the generator $\mathcal{L}_G$ are defined as follows:

$$\begin{aligned}\mathcal{L}_G =& \mathcal{L}_{\text{cont}}(v^f_{\text{s\_image}}, s_{\text{s\_sent}}) + \mathcal{L}_{\text{cont}}(v^f_{\text{image}}, s_{\text{sent}}) \\ &+ \mathcal{L}_{\text{cont}}(v^f_{\text{regions}}, s_{\text{words}}) + \mathcal{L}_{\text{cluster}}(X^f) + \mathcal{L}^G_{\text{GAN}},\end{aligned} \tag{3.8}$$

$$\mathcal{L}^G_{\text{GAN}} = -\frac{1}{L}\sum\nolimits_{i=1}^{L} \log\left(\mathbb{E}\left[\log(D(X^f_i))\right]\right), \tag{3.9}$$

where f represents features extracted from a synthetic sample.

3.2.5 Summary

In summary, our approach adopts both contrastive learning and clustering learning, enabling the discriminator to provide comprehensive training feedback to the generator. This allows the generator to maximize the mutual information between paired information and group instances with similar semantic meaning, ensuring a structured joint space where similar instances are grouped together within an appropriate range regardless of their modalities. As a result, we can use a single-GAN-based network to produce high-quality synthetic results.

3.3 Experiments

Table 3.1: Quantitative evaluation between different methods on the Pororo-SV and Abstract Scenes. For FID, FSD, and the number of parameters on Pororo-SV, lower is better; for text-image cosine similarity (Cosine), higher is better.

	Pororo-SV dataset			Abstract dataset			Number of Trainable Parameters	
Method	FID↓	FSD↓	Cosine↑	FID↓	FSD↓	Cosine↑	Generator	Discriminator
StoryGAN	78.64	94.53	0.22	135.16	55.80	3.59	47.0M	47.2M
CP-CSV	67.76	71.51	0.32	-	-	-	86.9M	70.9M
DUCO	95.17	171.70	0.08	142.34	49.16	3.95	53.2M	47.2M
VLC	94.30	122.07	0.21	-	-	-	54.5M	47.2M
Ours	**39.17**	**41.18**	**11.93**	**76.51**	**19.74**	**9.28**	**72.4M**	**10.4M**

We evaluate our approach by comparing it with other methods. We adopt StoryGAN (Li et al., 2019c), CP-CSV (Song et al., 2020), DUCO (Maharana et al., 2021), and VLC (Maharana and Bansal, 2021) as baselines. StoryGAN is a sequential conditional generative adversarial network, CP-CSV uses StoryGAN as backbone and adds the other segmentation mask generation pipeline to improve character consistency, and DUCO and VLC are built on top of StoryGAN as well, which rely on auxiliary video captioning networks to keep consistency. StoryGAN, CP-CSV, DUCO, and VLC all have two sets of generative adversarial networks, one for image generation, and one for story generation.

3.3.1 Implementation

We evaluate our approach at the resolution 64×64 on Pororo-SV and 256×256 on Abstract Scenes, as Abstract Scenes provides larger-scale ground-truth images. To work on images at the resolution 256×256, we repeat the same upsampling blocks in the generator and downsampling blocks in the discriminator for baseline methods. For implementing online clustering learning, the text encoder is a pre-trained bi-directional LSTM (Xu et al., 2018), and the image encoder is a VGG-16 (Simonyan and Zisserman, 2014), pre-trained on ImageNet. For offline clustering, we adopt the same text and image encoders, but train both using Eq. 3.5 from scratch, and freeze both during the training of the generative model. The image batch size is $N = 40$. To do clustering, we combine 10 batches of instances together, and the number of clusters k in K-means is set to 8. The hyperparameter is $\delta = 0.001$. The network is trained for 240 epochs on both Pororo-SV and Abstract Scenes. The Adam optimizer (Kingma and Ba, 2014)

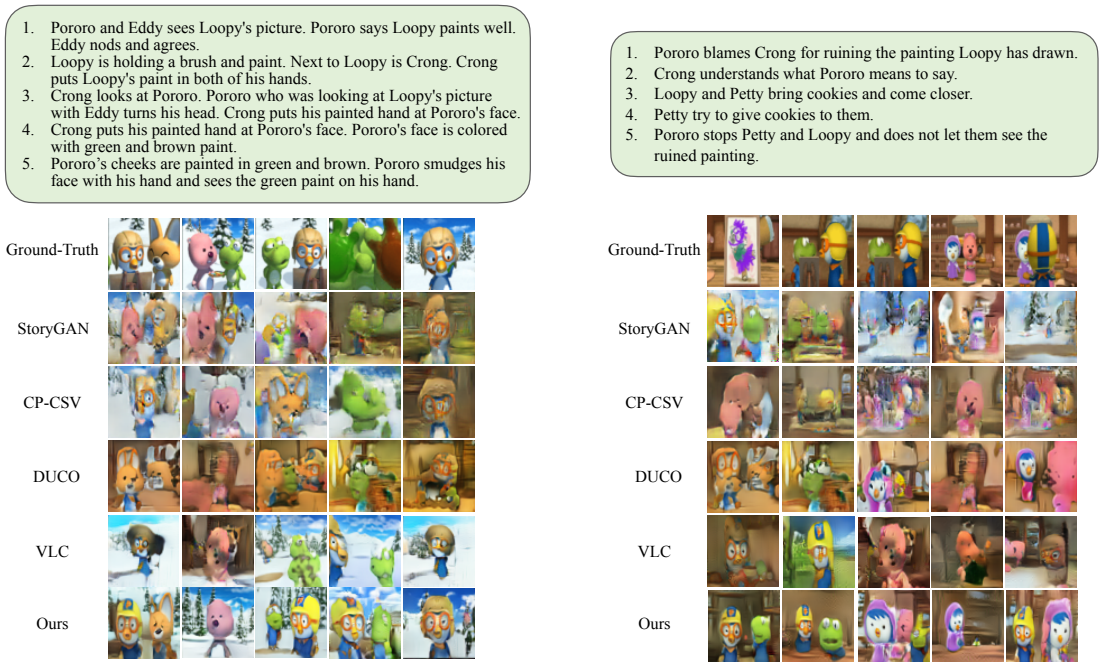


Figure 3.3: Qualitative comparison between methods on the Pororo-SV.

is adopted with learning rate 0.0002. We evaluate our approach and its variants on a single Quadro RTX 6000 GPU.

3.3.2 Quantitative Evaluation

We conduct a quantitative comparison between different methods on both Pororo-SV and Abstract, shown in Table 3.1. We find that our approach achieves the best results in all evaluation metrics on both datasets. Our approach improves the FID score from 78.64 to 39.17, and the FSD score from 94.53 to 41.18 on Pororo-SV, and creates a strong benchmark FID score of 76.51 and FSD score of 19.74 on Abstract, neither relying on segmentation masks nor complex video captioning auxiliary networks. Note that we do not evaluate CP-CSV and VLC on Abstract Scenes, as CP-CSV requires segmentation masks and VLC has not released the complete code.

Moreover, in Table 3.1, we show the number of parameters in the generator and the discriminator for different methods. Our approach has a smaller number of parameters. Compared to StoryGAN, the total number of parameters in the network reduces about 12.1%, especially reducing about 77.9% in the discriminator.

3.3.3 Qualitative Evaluation

In Figs. 3.3 and 3.4, we show the qualitative comparison between different methods on both Pororo-SV and Abstract. Although our approach has a single GAN with a

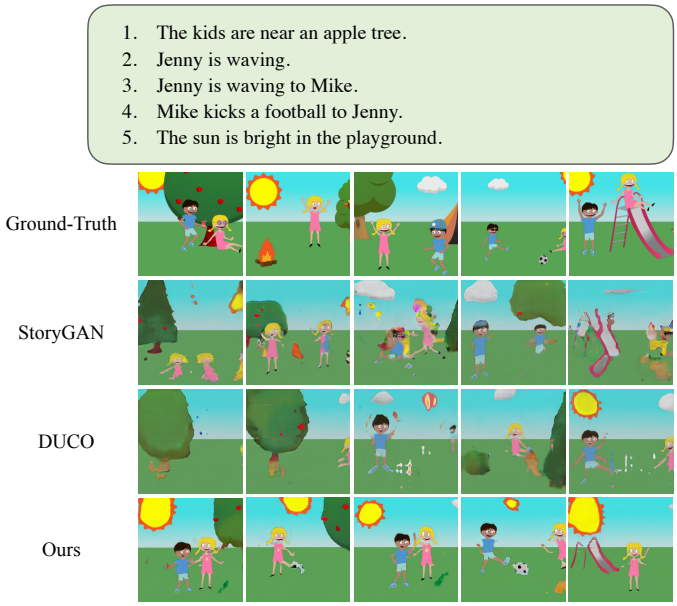


Figure 3.4: Qualitative comparison between methods on the Abstract Scenes.

fewer number of parameters, the synthetic story images have better image details: characters produced by our approach can be easily identified (e.g., penguin Pororo and pink bear Loopy) with clear outline and finer appearance (e.g., glasses and a hat on Pororo). As for Abstract, objects produced by our approach have a fine-grained shape (e.g., realistic characters boy Mike and girl Jenny), and ensure a good consistency between images within a story (e.g., the word sun only appears in the fifth sentence, but all images generated by our approach contain a sun in the sky).

3.3.4 Ablation Study

In Table 3.2, we evaluate different losses and analyze their impact on the Pororo-SV, including: (1) story image-story sentence (SI-SS), (2) image-sentence (I-S), (3) region-word (R-W), (4) story region-word (SR-W), and (5) clustering (cluster).

3.3.4.1 Individual Contrastive and Clustering Losses

Table 3.2 shows that using any of the contrastive and clustering losses improves all metrics compared to the baseline. The largest FID and FSD improvements come from the region-word contrastive loss, which improves FID from 96.54 to 50.09 and FSD from 103.72 to 67.17, respectively. However, using clustering loss alone would not significantly improve the performance. The ablations highlight the effectiveness of contrastive and clustering losses, where different contrastive losses maximize the

Table 3.2: Ablation results with different contrastive losses and clustering losses on the Pororo-SV, including: (1) story image-story sentence (SI-SS), (2) image-sentence (I-S), (3) region-word (R-W), (4) story region-word (SR-W), and (5) clustering (Cluster).

SI-SS	I-S	R-W	SR-W	Cluster	FID	FSD	Cosine
					96.54	103.72	1.53
✓					72.36	76.14	2.01
	✓				79.95	81.27	2.29
		✓			50.09	67.17	7.64
			✓		86.83	79.80	1.62
				✓	90.09	89.20	1.68
✓	✓	✓	✓		49.22	54.47	10.37
✓	✓	✓	✓	✓	39.17	41.18	11.93

mutual information at different representation levels, and the clustering loss helps capture semantic similarity and prevent contrastive learning pushing instances with similar semantic meaning far away from each other.

3.3.4.2 Combined Contrastive and Clustering Losses

Combining contrastive and clustering losses further improves all metrics. Using all contrastive losses achieves a better performance (FID 49.22 and FSD 54.47) than each individual. This demonstrates that different representation level contrastive losses are complementary, and using all of them can comprehensively explore mutual information between pairs to achieve the best improvement. Besides, contrastive and clustering losses also complement each other: the best FID and FSD scores come from combining all contrastive and clustering losses together, to obtain our final results.

3.3.4.3 Different Choices of Contrastive Methods

As discussed in Section 3.2.2, we suggest to adopt the masked margin softmax function (MMS) (Ilharco et al., 2019), instead of the normalized temperature-scaled cross-entropy loss (NT-Xent) (Chen et al., 2020), to define the similarity between representations from visual and language information and thus to exploit more additional negative. Here, we conduct a comparison study by replacing MMS with NT-Xent in all computation of contrastive losses, shown in Table 3.3. As we can see, using NT-Xent achieves worse scores on all metrics, degrading FID from 39.17 to 51.82, and FSD from 41.18 to 52.13. This indicates that comprehensively learning against a wider set of negative examples, namely, all those in the batch that are not known to be positively associated, can better explore mutual information across modalities, and thus improve image quality, text-image alignment, and story consistency.

Table 3.3: Comparison between different choices on the Pororo-SV.

Choice	FID	FSD
w/ NT-Xent	51.82	52.13
w/ Hierarchical Clustering	46.31	43.10
w/ Individual Feature	58.71	59.14
w/ Offline Learning	51.32	50.54
w/o BWSA	61.12	64.23
w/ WSA	50.13	53.25
4	186.77	282.61
8	39.17	41.18
16	43.95	42.46
32	42.87	45.29
64	45.34	45.83
Ours: MMS + K-means (8) + Fusion + Online + w/ BWSA	39.17	41.18

3.3.4.4 Different Choices of Clustering Methods

We evaluate different choices of clustering learning in our approach, including (1) selection of different clustering algorithms, such as hierarchical clustering and K-means, (2) inputs for clustering algorithms, i.e., fusion features and individual features from their own modality, and (3) online and offline clustering learning. As shown in Table 3.3, first, there were no significant changes on all metrics when using different clustering algorithms, so the choice of different clustering algorithms depends on the specific requirements of the task, as well as the strengths and weaknesses of each algorithm. Second, using individual features degraded the performance of our method, reflected by worse FID and FSD. This indicates that using fusion features encourages each modality feature to move closer to the semantic centroid, by explicitly encouraging semantically close features from different domains to cluster together. Third, online clustering learning achieved a better performance compared to offline learning, reflected by improvements on all metrics, which means that the training prediction heads along with generative models could better convert different modality features into a joint space to achieve better translation from language inputs into visual contexts.

3.3.4.5 Number of Clusters

Table 3.3 shows the results using different number of cluster sizes for K-means. As we can see, results considerably improve when the number of clusters increases from 4

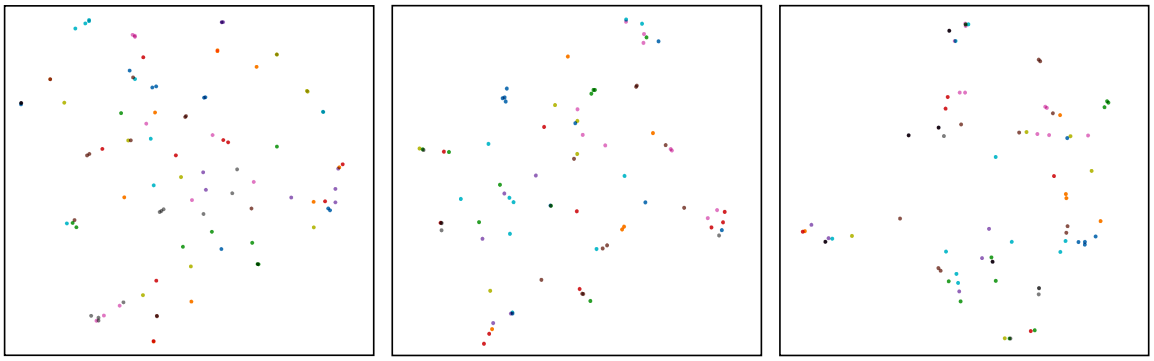


Figure 3.5: t-SNE plots of effects of clustering learning. Images from the same story have the same color.

to 8, and become stable after 8 on Pororo-SV. Based on this, we set the number of clusters $k = 8$ for our clustering algorithm.

3.3.4.6 Visualization of Effects of Clustering Learning

We visualize synthetic story images on Pororo-SV using t-SNE plots, shown in Fig. 3.5. For this plot, we randomly choose 16 story sentences and feed them into three methods separately, to generate 16 stories with 80 images for each method. Then, we use a pre-trained VGG-16 to extract a global image feature from each image with dimension $\mathbb{R}^{356}$ and visualize these features using t-SNE. Basically, images within the same story are close to each other, as they have a certain semantic similarity, and images from different stories may be close if they have similar semantic meaning. We observe that with the help of contrastive learning only (middle), semantically related instances (e.g., instances from the same story) tend to be more tightly related than without using both learning methods (left), and using clustering learning (right) can further pull similar instances together than using contrastive learning only. Also, image features from different stories are clearly more separable using clustering learning, if the stories have a low semantic similarity.

3.3.4.7 Bi-Directional Attention

By Table 3.3, without BWSA, the evaluation results on metrics degrade. This demonstrates that BWSA is complementary to the discriminator, where the discriminator uses information from story-level to word-level to provide comprehensive training feedback to the generator, and under the guidance of such fine-grained feedback, the generator uses BWSA to capture relations between language and visual information.

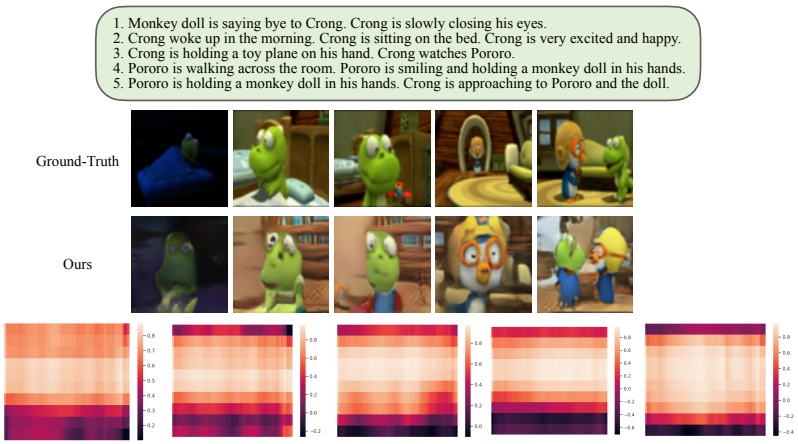


Figure 3.6: Heat map of bi-directional attention.

Furthermore, we replace BWSA with word-level spatial attention (Xu et al., 2018) (WSA) to verify its effect. Compared to w/o BWSA, w/ WSA improves all scores, but the scores are still worse than the full model, which indicates that the negative relation in our proposed BWSA can further improve the output results. We attribute this improvement to the comprehensive capture of different relations between words and image regions, instead of the positive effects that are only contained in WSA. We think that our proposed BWSA is a complete version of word-level spatial attention that captures both positive and negative relations between image regions and different words, where a positive effect of BWSA works similarly to WSA to highlight image regions aligned with corresponding semantic words, and a negative effect of BWSA works as a complementary component to capture relations that cannot be observed by the positive part.

3.3.5 Visualization of Bi-Directional Attention

In Fig. 3.6, we show the heat map of the relation matrix $\alpha \in \mathbb{R}^{(H*W)\times L}$ (shown in Fig. 3.2, red region) to visualize both positive and negative effects between image regions and words that are captured by our proposed BWSA. The bottom is the heat map of relations between image regions and words, corresponding to each synthetic story image. We extract the relation matrix α at the resolution 64×64. However, there are a total of 4096 spatial locations in the hidden features, which is not good for visualization. So, we use 64 image regions to represent the whole spatial locations by computing the average in each region. Also, the number of words varies, and thus we choose the first 10 words for visualization. Thus, the y and the x axis of each heat map represent 10 different words and 64 different image regions, respectively. As we

can see, the values in the heat map range from -1 to 1, instead of ≥ 0, and dark colors (< 0) appear in multiple locations, which indicates that there exist both positive and negative relations between words and image regions, and thus to comprehensively capture all relations can further improve the generation of story images.

3.3.6 Human Evaluation

Table 3.4: Human evaluation on the Pororo-SV based on three criteria.

Choice (%)	Ours	VLC	DUCO
Visual Quality	78.67	11.33	10.00
Alignment	73.67	16.00	10.33
Consistency	82.00	9.67	8.33

A human evaluation study on Pororo-SV is conducted to further evaluate the effectiveness of our approach. Following StoryGAN (Li et al., 2019c), three evaluation criteria are chosen: (1) visual quality, (2) text-image semantic alignment, and (3) consistency across story images. In this human evaluation, we ask workers to decide which sample is the best, where each sample contains story images and corresponding sentences. 100 randomly selected samples are assigned to three workers to reduce the human variance. As shown in Table 3.4, workers prefer the synthetic story images that are generated by our approach on these three evaluation criteria.

3.4 Conclusion

We studied how to use both contrastive learning and clustering learning to maximize mutual information between instances with the similar semantic meaning to ensure a structured joint space. Besides, we introduced fusion features and use the features in both clustering learning and the discriminator, and further extended word-level spatial attention to comprehensively capture both positive and negative relations between words and image regions. By doing this, we have a concise single-GAN-based network, based on contrastive and clustering learning, where experimental results demonstrate the superior performance of our proposed approach, even if it has a small number of parameters compared to the baselines.

Chapter 4

Build Connections between Different-Modal Information

In this chapter, I mainly focus on exploring different approaches to achieve an effective interaction between multi-modal information. To achieve this, three approaches are proposed: (1) word-level attention-based connection, helped with a complementary word-level discriminator, (2) a text-image affine combination module, and (3) a semi-parametric memory-driven approach, where all approaches can allow the network to learn to effectively correlate image regional visual attributes with corresponding semantic words from a given sentence. To evaluate our proposed approaches, two tasks are adopted: text-to-image generation and text-guided image manipulation. More details about two tasks are described in Chapter 2.

4.1 Word-Level Attention-Based Connection with a Complementary Discriminator

Basically, it is challenging to generate image content that matches the textual description due to the domain gap between these two modalities. To address this challenge, we propose to utilize word-level attention to enable the network to learn how to correlate image visual attributes with semantically aligned words from a given sentence. Additionally, we introduce a complementary word-level discriminator to provide fine-grained training feedback to the generator, helping it to capture the cross-domain relationships accurately. In the following sections, we first provide the background of the task used to evaluate our proposed approaches, followed by more details on how to use attention to effectively correlate information from different modalities.

4.1.1 Background of Controllability in Text-to-Image Generation

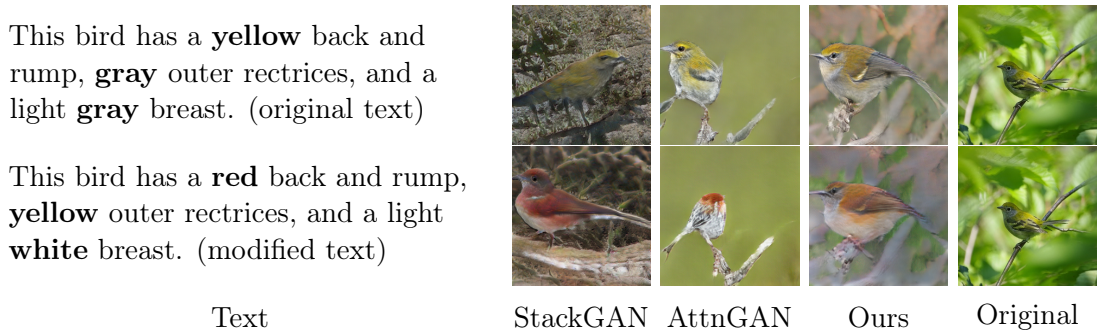


Figure 4.1: Examples of modifying synthetic images by editing the natural language description. The current methods can generate realistic images, but fail to modify the synthetic image by changing the given sentence. In contrast, our method allows parts of the image to be manipulated in correspondence to the modified text description while preserving other unrelated content.

Generating realistic images that semantically match a given text descriptions is a challenging problem and has tremendous potential applications, such as image editing, video games, and computer-aided design. Recently, thanks to the success of generative adversarial networks (GANs) (Denton et al., 2015; Goodfellow et al., 2014; Radford et al., 2015) in generating realistic images, text-to-image generation has made remarkable progress (Reed et al., 2016b; Xu et al., 2018; Zhang et al., 2017a) by implementing conditional GANs (cGANs) (Dong et al., 2017; Reed et al., 2016b,a), which are able to generate realistic images conditioned on given text descriptions.

However, current generative networks are typically uncontrollable, which means that if users change some words of a sentence, the synthetic image would be significantly different from the one generated from the original text as shown in Fig. 4.1. When the given text description (e.g., color) is changed, corresponding visual attributes of the bird are modified, but other unrelated attributes (e.g., the pose and position) are changed as well. This is typically undesirable in real-world applications, when a user wants to further modify the synthetic image to satisfy her preferences.

So, the goal of this paper is to generate images from text, and also allow the user to manipulate synthetic images using natural language descriptions, in one framework. In particular, we focus on modifying visual attributes (e.g., category, texture, and color) of objects in the generated images by changing given text descriptions.

To solve this problem, we propose an attention-based approach to effectively correlate image visual attributes with corresponding semantic words from a given

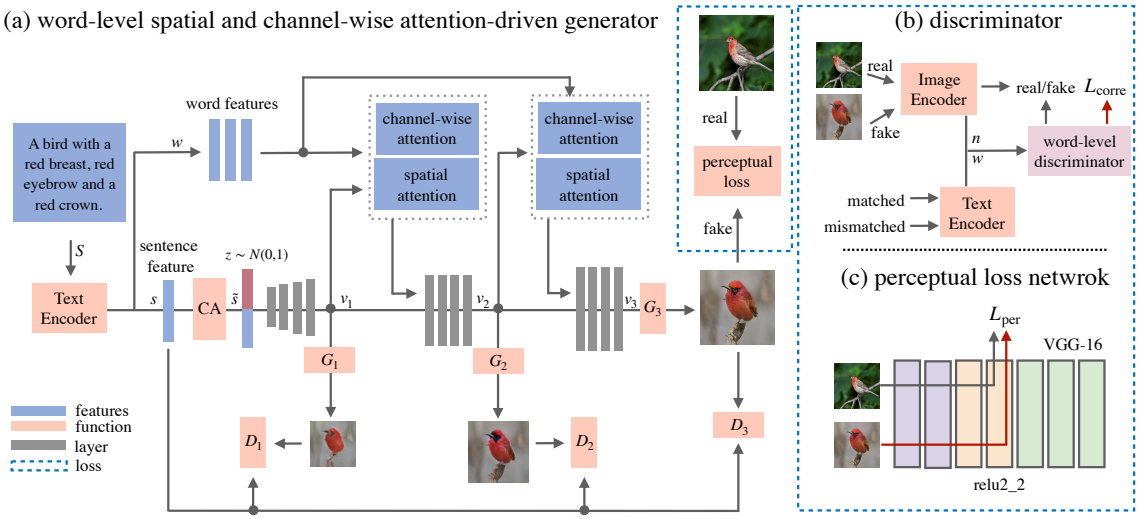


Figure 4.2: Architecture of our proposed ControlGAN. In (b), $\mathcal{L}_{\text{corre}}$ is the correlation loss. In (c), $\mathcal{L}_{\text{per}}$ is the perceptual loss.

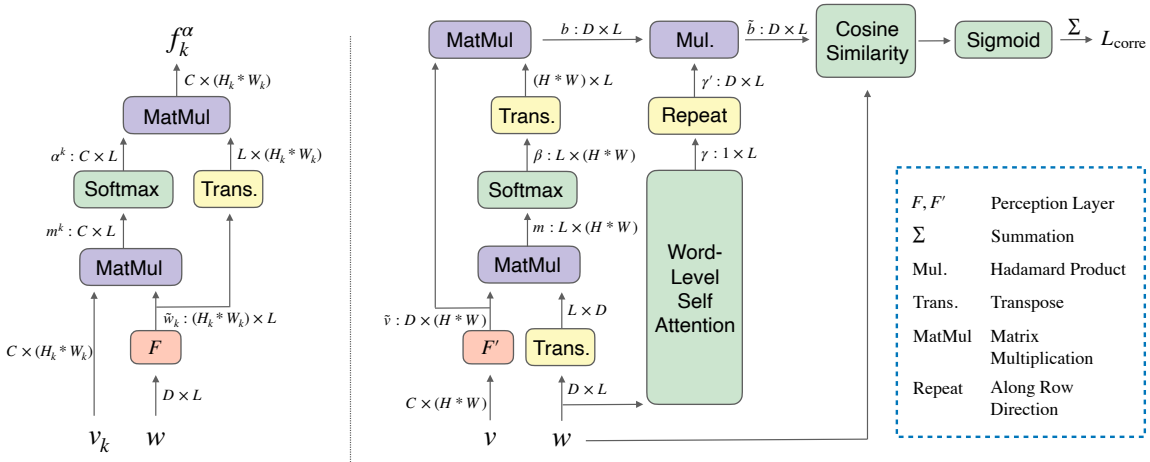
sentence, where attention allows the network to learn to identify specific image visual attributes, semantically aligned with the corresponding words, and thus the network can only modify specific visual attributes in the synthetic images by changing corresponding words of a given sentence. Meanwhile, a complementary word-level discriminator is proposed to help the generator to correctly capture this relations. More details are shown as follow.

4.1.2 Attention-Based Controllable Text-to-Image Generation

The complete multi-stage architecture is shown in Fig. 4.2, including generators with word-level spatial and channel-wise attention-based connection, word-level discriminators, and a perceptual loss network. In the following, the term "channel" refers to individual feature maps.

4.1.2.1 Attention-Based Connection

The generator in our proposed network exploits the attention mechanism via incorporating a spatial attention module (Xu et al., 2018) and the proposed channel-wise attention module. The spatial attention module can only correlate words with individual spatial locations without taking channel information into account. Thus, we introduce a channel-wise attention module to exploit the connection between words and channels. We experimentally find that the channel-wise attention module highly correlates semantically meaningful parts of the image with corresponding words,



(a) channel-wise attention (b) word-level discriminator

Figure 4.3: Architecture of the proposed channel-wise attention module and word-level discriminator, where the "channel" refers to individual feature maps of hidden features.

while the spatial attention focuses on color descriptions (see Fig. 4.6). Therefore, our proposed channel-wise attention module, together with the spatial attention, can help the generator disentangle different visual attributes, and allow it to focus only on the most relevant subregions and channels.

At the k^{th} stage, the channel-wise attention module (see Fig. 4.3 (a)) takes two inputs: the word features $w \in \mathbb{R}^{D \times L}$ and hidden visual features $v_k \in \mathbb{R}^{C \times (H_k * W_k)}$, where D represents the dimensionality of the word embedding space, L is the number of words, H_k and W_k denote the height and width of the feature map at stage k, respectively, and C is the number of channels. The word embeddings w are first mapped into the same semantic space as the visual features v_k via a perception layer F, producing $\tilde{w}_k = Fw$, where $F \in \mathbb{R}^{(H_k * W_k) \times D}$.

Then, we calculate the channel-wise attention matrix $m^k \in \mathbb{R}^{C \times L}$ by multiplying the converted word features $\tilde{w}_k$ and visual features v_k, denoted as $m^k = v_k \tilde{w}_k$. Thus, m^k aggregates correlation values between channels and words across all spatial locations, where each element in m^k is the sum of correlation values at all spatial locations. Next, m^k is normalized by the softmax to generate the normalized channel-wise attention matrix α^k as

$$\alpha_{i,j}^k = \frac{\exp(m_{i,j}^k)}{\sum_{l=0}^{L-1} \exp(m_{i,l}^k)}. \tag{4.1}$$

The attention weight $\alpha_{i,j}^k$ represents the correlation between the i^{th} channel in the visual features v_k and the j^{th} word in the sentence S, and higher value means larger correlation between them.

Equipped with the channel-wise attention matrix α^k, we obtain the final channel-wise attention features $f_k^\alpha \in \mathbb{R}^{C\times(H_k*W_k)}$, denoted as $f_k^\alpha = \alpha^k(\tilde{w}_k)^T$. Thus, channels with high correlation values are enhanced resulting in a high response to corresponding words, which can facilitate disentangling word attributes into different channels, and also reduce the influence from irrelevant channels by assigning a lower correlation.

4.1.2.2 Word-Level Discriminator

To encourage the generator to modify only parts of the image according to the text, the discriminator should provide the generator with fine-grained training feedback, which can guide the generation of subregions corresponding to the most relevant words. Actually, the text-adaptive discriminator (Nam et al., 2018) also explores the word-level information in the discriminator, but it adopts a global average pooling layer to output a 1D vector as image feature, and then calculates the correlation between image feature and each word. By doing this, the image feature may lose important spatial information, which provides crucial cues for disentangling different visual attributes. To address the issue, we propose a novel word-level discriminator to explore the correlation between image subregions and each word; see Fig. 4.3 (b).

Our word-level discriminator takes two inputs: 1) word features w, w' encoded from the text encoder, which follows the same architecture as the one (see Fig. 4.2 (a)) used in the generator, where w and w' denote word features encoded from the original text S and a randomly sampled mismatched text, respectively, and 2) visual features $v_{\text{real}}, v_{\text{fake}}$, both encoded by a GoogleNet-based (Szegedy et al., 2015) image encoder from the real image I and generated images I', respectively.

For simplicity, we use $v \in \mathbb{R}^{C\times(H*W)}$ to represent visual features v_{real} and v_{fake} at stage k, and use $w \in \mathbb{R}^{D\times L}$ for both original and mismatched word features. The word-level discriminator contains a perception layer F' that is used to align the channel dimension of visual feature v and word feature w, denoted as $\tilde{v} = F'v$, where $F' \in \mathbb{R}^{D\times C}$ is a weight matrix to learn. The reason for performing alignment in the word space rather than in the visual space is that word-level self-attention will be further utilized to update image subregion-aware word features b in the subsequent steps. The word-context correlation matrix $m \in \mathbb{R}^{L\times(H*W)}$ is derived via $m = w^T\tilde{v}$, and is further normalized by the softmax function to get a correlation matrix β:

$$\beta_{i,j} = \frac{\exp(m_{i,j})}{\sum_{l=0}^{(H*W)-1}\exp(m_{i,l})}, \tag{4.2}$$

where $\beta_{i,j}$ represents the correlation value between the i^{th} word and the j^{th} subregion of the image. Then, the image subregion-aware word features $b \in \mathbb{R}^{D\times L}$ can be obtained

by $b = \tilde{v}\beta^T$, which aggregates all spatial information weighted by the word-context correlation matrix β.

Additionally, to further reduce the negative impact of less important words, we adopt the word-level self-attention (Nam et al., 2018) to derive a 1D vector γ with length L reflecting the relative importance of each word. Then, we repeat γ by D times to produce γ', which has the same size as b. Next, b is further reweighted by γ' to get $\tilde{b}$, denoted as $\tilde{b} = b \odot \gamma'$, where $\odot$ represents element-wise multiplication. Finally, we derive the correlation between the i^{th} word and the whole image as Eq. (4.3):

$$r_i = \sigma(\frac{(\tilde{b}_i)^T w_i}{||\tilde{b}_i||\ ||w_i||}), \tag{4.3}$$

where σ is the sigmoid function, r_i evaluates the correlation between the i^{th} word and the image, and $\tilde{b}_i$ and w_i represent the i^{th} column of b and w, respectively.

Therefore, the final correlation value $\mathcal{L}_{\text{corre}}$ between image I and sentence S is calculated by summing all word-context correlations, denoted as $\mathcal{L}_{\text{corre}}(I, S) = \sum_{i=0}^{L-1} r_i$. By doing so, the generator can receive fine-grained feedback from the word-level discriminator for each visual attribute, which can further help supervise the generation and manipulation of each subregion independently.

4.1.2.3 Perceptual Loss

Without adding any constraint on text-irrelevant regions (e.g., backgrounds), the generated results can be highly random, and may also fail to be semantically consistent with other content. To mitigate this randomness, we adopt the perceptual loss (Johnson et al., 2016) based on a 16-layer VGG network (Simonyan and Zisserman, 2014) pre-trained on the ImageNet dataset (Russakovsky et al., 2015). The network is used to extract semantic features from both the generated image I' and the real image I, and the perceptual loss is defined as

$$\mathcal{L}_{\text{per}}(I', I) = \frac{1}{C_i H_i W_i} \left\| \phi_i(I') - \phi_i(I) \right\|_2^2, \tag{4.4}$$

where $\phi_i(I)$ is the activation of the i^{th} layer of the VGG network, and H_i and W_i are the height and width of the feature map, respectively. The intuition behind perceptual loss is to leverage pre-trained deep neural networks to capture and compare higher-level visual features, allowing the learning process to be guided towards generating visually plausible and semantically aligned images that closely resemble the corresponding ground-truth images.

4.1.2.4 Objective Functions

The generator and discriminator are trained alternatively by minimizing both the generator loss $\mathcal{L}_G$ and discriminator loss $\mathcal{L}_D$.

Generator objective. The generator loss $\mathcal{L}_G$ as Eq. (4.5) contains an adversarial loss $\mathcal{L}_{G_k}$, a text-image correlation loss $\mathcal{L}_{\text{corre}}$, a perceptual loss $\mathcal{L}_{\text{per}}$, and a text-image matching loss $\mathcal{L}_{\text{DAMSM}}$ (Xu et al., 2018).

$$\mathcal{L}_G = \sum_{k=1}^{K}(\mathcal{L}_{G_k} + \lambda_2 \mathcal{L}_{\text{per}}(I_k{}', I_k) + \lambda_3 \log(1 - \mathcal{L}_{\text{corre}}(I_k', S))) + \lambda_4 \mathcal{L}_{\text{DAMSM}}, \tag{4.5}$$

where K is the number of stages, I_k is the real image sampled from the true image distribution P_{data} at stage k, $I_k{}'$ is the generated image at the k^{th} stage sampled from the model distribution PG_k, $\lambda_2, \lambda_3, \lambda_4$ are hyper-parameters controlling different losses, $\mathcal{L}_{\text{per}}$ is the perceptual loss described in Section 4.1.2.3, which puts constraint on the generation process to reduce the randomness, the $\mathcal{L}_{\text{DAMSM}}$ (Xu et al., 2018) is used to measure text-image matching score based on the cosine similarity, and $\mathcal{L}_{\text{corre}}$ reflects the correlation between the generated image and the given text description considering spatial information.

The adversarial loss $\mathcal{L}_{G_k}$ is composed of the unconditional and conditional adversarial losses shown in Eq. (4.6): the unconditional adversarial loss is applied to make the synthetic image be real, and the conditional adversarial loss is utilized to make the generated image I' match the given text S.

$$\mathcal{L}_{G_k} = \underbrace{-\frac{1}{2} E_{I_k{}' \sim PG_k}\left[\log(D_k(I_k{}'))\right]}_{\text{unconditional adversarial loss}} \underbrace{-\frac{1}{2} E_{I_k{}' \sim PG_k}\left[\log(D_k(I_k{}', S))\right]}_{\text{conditional adversarial loss}}. \tag{4.6}$$

Discriminator objective. The final loss function for training the discriminator D is defined as:

$$\mathcal{L}_D = \sum_{k=1}^{K}(\mathcal{L}_{D_k} + \lambda_1(\log(1 - \mathcal{L}_{\text{corre}}(I_k, S)) + \log \mathcal{L}_{\text{corre}}(I_k, S'))), \tag{4.7}$$

where $\mathcal{L}_{\text{corre}}$ is the correlation loss determining whether word-related visual attributes exist in the image (see Section 4.1.2.2), S' is a mismatched text description that is randomly sampled from the dataset and is irrelevant to I_k, and λ_1 is a hyper-parameter controlling the importance of additional losses.

The adversarial loss $\mathcal{L}_{D_k}$ contains two components: the unconditional adversarial loss determines whether the image is real, and the conditional adversarial loss

determines whether the given image (I or I') matches the text description S:

$$\begin{aligned}\mathcal{L}_{D_k} = & \underbrace{-\frac{1}{2}E_{I_k\sim P_{\text{data}}}\left[\log(D_k(I_k))\right] - \frac{1}{2}E_{I_k{}'\sim PG_k}\left[\log(1-D_k(I_k{}'))\right]}_{\text{unconditional adversarial loss}} \\ & \underbrace{-\frac{1}{2}E_{I_k\sim P_{\text{data}}}\left[\log(D_k(I_k,S))\right] - \frac{1}{2}E_{I_k{}'\sim PG_k}\left[\log(1-D_k(I_k{}',S))\right]}_{\text{conditional adversarial loss}}.\end{aligned} \tag{4.8}$$

4.1.2.5 Summary

In this chapter, we propose to use word-level attention to correlate image visual attributes with corresponding semantic words from a given sentence, helped with fine-grained training feedback provided by a complementary word-level discriminator. By doing this, our network effectively disentangle different attributes and semantic words, and build an accurate connection between them to achieve an effective interaction.

4.1.3 Experiments

Table 4.1: Quantitative comparison: Inception score, R-precision, and L_2 reconstruction error of baselines and ControlGAN on the CUB and COCO.

	CUB			COCO		
Method	IS	Top-1 Acc(%)	L_2 error	IS	Top-1 Acc(%)	L_2 error
StackGAN++	4.04 ± .05	45.28 ± 3.72	0.29	8.30 ± .10	72.83 ± 3.17	0.32
AttnGAN	4.36 ± .03	67.82 ± 4.43	0.26	**25.89 ± .47**	**85.47 ± 3.69**	0.40
Ours	**4.58 ± .09**	**69.33 ± 3.23**	**0.18**	24.06 ± .60	82.43 ± 2.43	**0.17**

To evaluate the effectiveness of our approach, we conduct extensive experiments on the CUB bird (Wah et al., 2011) and the MS COCO (Lin et al., 2014) datasets. We compare with two GAN methods on text-to-image generation, StackGAN++ (Zhang et al., 2018a) and AttnGAN (Xu et al., 2018). Results for the baselines are reproduced based on the code released by the authors.

4.1.3.1 Implementation

There are three stages ($K = 3$) in our ControlGAN generator following Xu et al. (2018). The three scales are 64×64, 128×128, and 256×256, and spatial and channel-wise attentions are applied at the stages 2 and 3. The text encoder is a pre-trained bidirectional LSTM (Schuster and Paliwal, 1997) to encode the given text description into a sentence feature with dimension 256 and word features with length

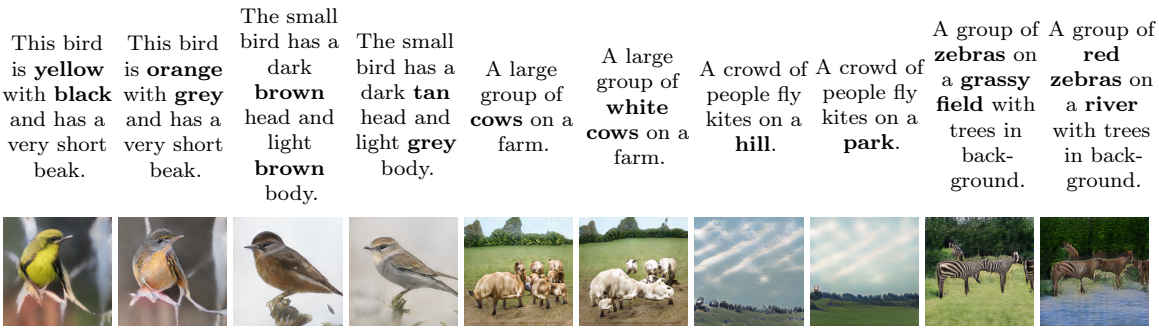


Figure 4.4: Qualitative results on the CUB and COCO. Odd-numbered columns show the original text and even-numbered ones the modified text. The last two are an out-of-distribution case.

18 and dimension 256. In the perceptual loss, we compute the content loss at layer relu2_2 of VGG-16 (Simonyan and Zisserman, 2014) pre-trained on the ImageNet (Russakovsky et al., 2015). The whole network is trained using the Adam optimizer (Kingma and Ba, 2014) with the learning rate 0.0002. The hyper-parameters λ_1, λ_2, λ_3, and λ_4 are set to 0.5, 1, 1, and 5 for both datasets, respectively.

4.1.3.2 Comparison with State of the Art

Quantitative results. We adopt the Inception score (Salimans et al., 2016) to evaluate the quality and diversity of the generated images. However, as the Inception score cannot reflect the relevance between an image and a text description, we utilize R-precision (Xu et al., 2018) to measure the correlation between a generated image and its corresponding text. We compare the top-1 text-to-image retrieval accuracy (Top-1 Acc) on the CUB and COCO datasets following Nam et al. (2018).

Quantitative results are presented in Table 4.1. Through random repeated runs, our method achieves higher IS and R-precision values on the CUB dataset compared to the state of the art, indicating the generation of higher-quality images with improved diversity that semantically align with the text descriptions. Additionally, our method demonstrates competitive performance on the COCO dataset.

To evaluate the controllability of the generated results, the L_2 reconstruction error (Nam et al., 2018) between the images generated from the original and modified texts is computed in Table 4.1. ControlGAN achieves a significantly lower reconstruction error compared to other methods, indicating that our approach better preserves the content in the image generated from the original text.

Qualitative results. We present qualitative comparisons in Fig. 4.4. Our approach showcases the ability to accurately manipulate specific visual attributes

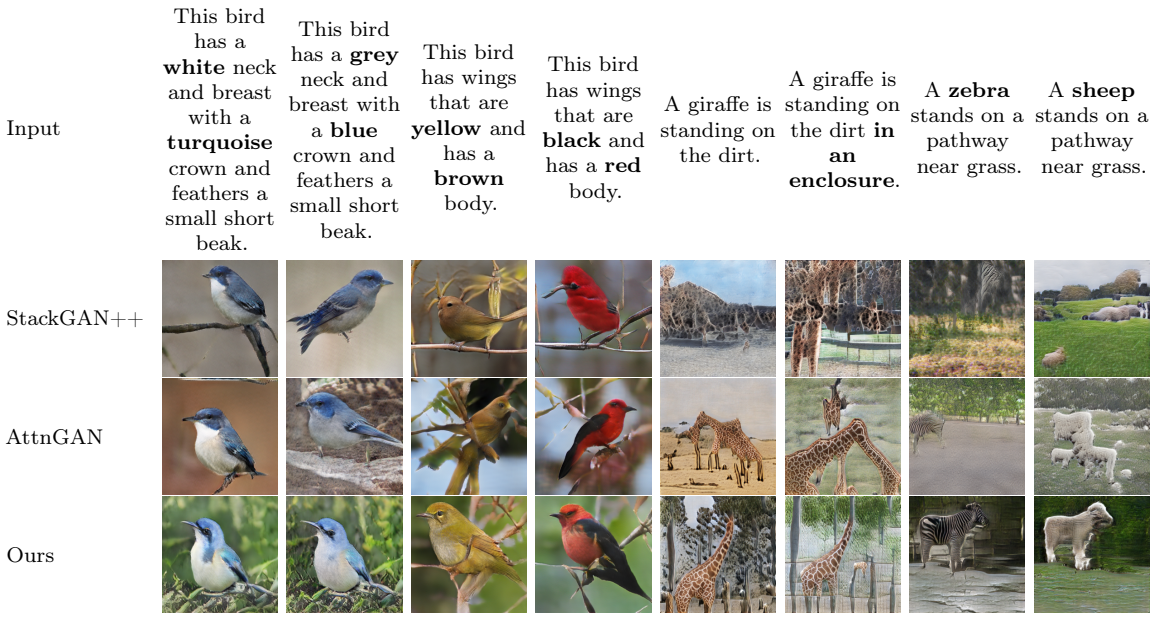


Figure 4.5: Qualitative comparison of methods on the CUB and COCO. Odd-numbered columns show the original text and even-numbered ones the modified text.

based on the given text descriptions. Moreover, our method demonstrates its capability in handling out-of-distribution text-to-image generation, as depicted in the last two columns of Fig. 4.4 where a group of non-existent red zebras on a river is generated. These results underscore the effectiveness of our approach in disentangling visual attributes for text-to-image generation, enabling independent manipulation of different visual attributes.

Fig. 4.5 shows the visual comparison between ControlGAN, AttnGAN (Xu et al., 2018), and StackGAN++ (Zhang et al., 2018a). It can be observed that when the text is modified, the two compared approaches are more likely to generate new content, or change some visual attributes that are not relevant to the modified text. For instance, as shown in the first two columns, when we modify the color attributes, StackGAN++ changes the pose of the bird, and AttnGAN generates new background. In contrast, our approach is able to accurately manipulate parts of the image generation corresponding to the modified text, while preserving the visual attributes related to unchanged text.

In COCO, our model again achieves much better results compared with others shown in Fig. 4.5. For example, as shown in the last four columns, the compared approaches cannot preserve the shape of objects and even fail to generate reasonable images. Generally speaking, the results on COCO are not as good as on the CUB dataset. We attribute this to the few text-image pairs and more abstract captions in the dataset. Although there are a lot of categories in COCO, each category only has a

few number of examples, and captions focus mainly on the category of objects rather than detailed descriptions, which makes text-to-image generation more challenging.

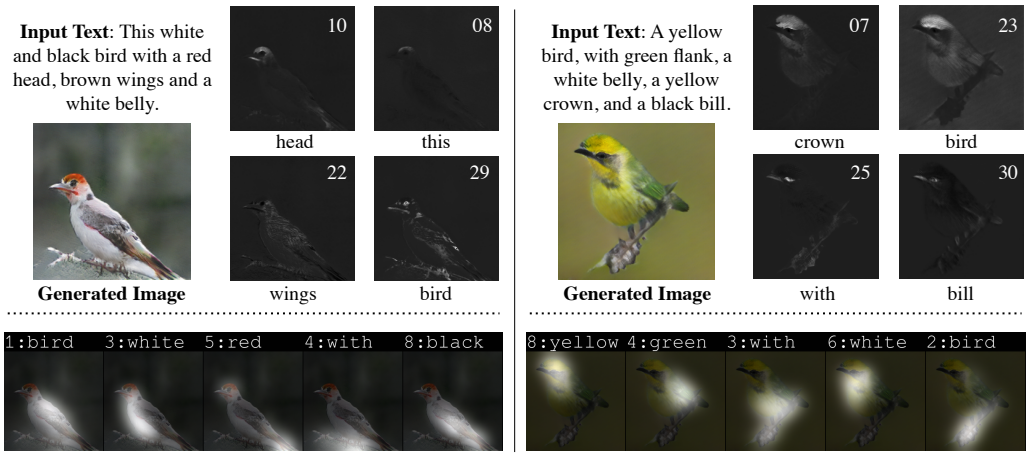


Figure 4.6: Top: Visualization of feature channels at stage 3. The number at the top-right corner indicates the channel number, and the word with the highest correlation $\alpha_{i,j}$ in Eq. 4.1 with the channel is displayed below the image. Bottom: Spatial attention produced in stage 3.

4.1.3.3 Component Analysis

Effectiveness of channel-wise attention. Our model implements channel-wise attention in the generator, together the spatial attention, to generate realistic images. To better understand the effectiveness of attention mechanisms, we visualize the intermediate results and corresponding attention maps at different stages.

We have experimentally observed that the channel-wise attention module closely correlates with semantic parts of objects, while the spatial attention module focuses mainly on color descriptions. Fig. 4.6 shows several channels of feature maps that correspond to different semantics, and our channel-wise attention module assigns large correlation values to channels that are semantically related to the words describing parts of a bird, such as "hear", "wings", "bill", and "crown". This phenomenon is further confirmed by the ablation study shown in Fig. 4.7 (left side). Without the channel-wise attention module, our model fails to generate controllable results when we modify the text related to parts of a bird. In contrast, our model with the channel-wise attention module can generate better controllable results.

Effectiveness of word-level discriminator. To verify the effectiveness of the word-level discriminator, we first conduct an ablation study: our model is trained without word-level discriminator, shown in Fig. 4.7 (right side), and then we construct

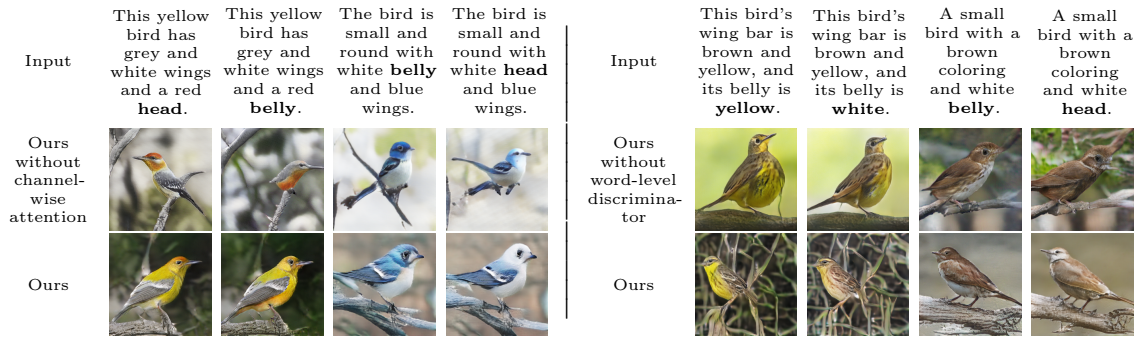


Figure 4.7: Left: Ablation study of channel-wise attention. Right: Ablation study of the word-level discriminator. Without either component, our model fails to maintain consistency between the new synthetic image and its initial synthetic image (e.g., different pose and shape) when modifying the text related to parts of a bird.

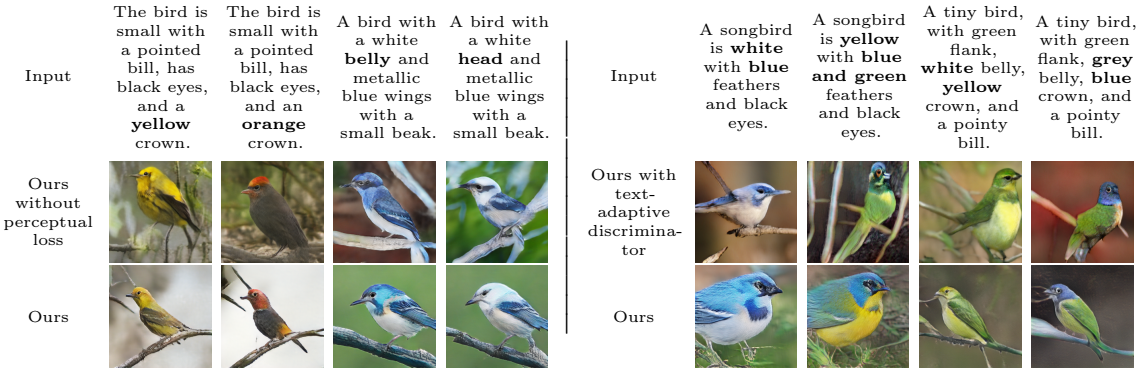


Figure 4.8: Left: Ablation study of the perceptual loss. Right: Comparison between our word-level discriminator and the text-adaptive discriminator (Nam et al., 2018). From the comparison, we can observe that the perceptual loss helps preserve text-irrelevant contents, and our network can better modify image parts due to the fine-grained training feedback provided by our proposed discriminator.

a baseline model by replacing our discriminator with a text-adaptive discriminator (Nam et al., 2018), which also explores the correlation between image features and words. Visual comparisons are shown in Fig. 4.8 (right side). We can easily observe that the compared baseline fails to manipulate the synthetic images. For example, as shown in the first two columns, the bird generated from the modified text has a totally different shape, and the background has been changed as well. This is due to the fact that the text-adaptive discriminator (Nam et al., 2018) uses a global pooling layer to extract image features, which may lose important spatial information.

Effectiveness of perceptual loss. We conducted an ablation study where our model was trained without the perceptual loss, as shown in Fig. 4.8 (left side). Without the perceptual loss, the generated images from modified text struggle to

preserve content related to the unmodified text. This indicates that the perceptual loss introduces a stricter semantic constraint on image generation and helps reduce the level of randomness involved.

4.1.4 Conclusion

We have proposed a controllable generative adversarial network (ControlGAN), which can generate and manipulate the generation of images based on natural language descriptions. Our ControlGAN can successfully disentangle different visual attributes and allow parts of the synthetic image to be manipulated accurately, while preserving the generation of other content. Three novel components are introduced in our model: (1) the word-level spatial and channel-wise attention-driven generator can effectively disentangle different visual attributes, (2) the word-level discriminator provides the generator with fine-grained training signals related to each visual attribute, and (3) the adoption of perceptual loss reduces the randomness involved in the generation, and enforces the generator to reconstruct content related to unmodified text. Extensive experimental results demonstrate the effectiveness and superiority of our method on two benchmark datasets.

4.2 Text-Image Affine Combination

In this section, we introduce a novel approach for combining text and image information to enable effective interaction between the two modalities. Unlike attention mechanisms, which can be computationally expensive and overlook text-irrelevant image features, our proposed method, called the text-image affine combination module, offers a simple yet effective solution. It leverages an affine transformation to merge text and image features during the generation process, allowing the network to selectively integrate text-required image regional information into the generation pipeline while preserving text-irrelevant image contents.

In the following, we provide the background of the task used to evaluate our approaches and delve into the details of how the affine transformation is utilized to achieve an efficient and meaningful interaction between text and image information.

4.2.1 Background of Text-Guided Image Manipulation

Image manipulation aims to modify some aspects of a given image, from low-level texture or color (Gatys et al., 2016; Zhang et al., 2016) to high-level semantics (Zhu et al., 2016), to meet a user's preferences, which has numerous potential applications in video games, image editing, and computer-aided design. At the time of writing this paper, with the development of deep learning and deep generative models, automatic image manipulation has made remarkable progress, including image inpainting (Iizuka et al., 2016; Pathak et al., 2016), image colorization (Zhang et al., 2016), style transfer (Gatys et al., 2016; Johnson et al., 2016), and domain or attribute translation (Isola et al., 2017; Lample et al., 2017).

All of the aforementioned studies primarily focus on specific problems, and there are only a few studies that address the broader and more user-friendly domain of text-guided image manipulation. Specifically, text-guided image manipulation aims to semantically edit specific parts of an image based on natural language descriptions provided by the user, while preserving the rest of the image content that is not mentioned in the text. However, at the time of writing this paper, the existing state-of-the-art text-guided image manipulation methods, such as those proposed by Dong et al. (2017) and Nam et al. (2018), have limitations in generating high-quality images. As shown in Fig. 4.9, the first row depicts low-quality results produced by these methods, which are far from satisfactory. Moreover, these methods often struggle to effectively manipulate complex scenes, as illustrated in the second row of Fig. 4.9.

To achieve an effective image manipulation guided by text descriptions, the key is to exploit both text and image cross-modality information, generating new attributes matching the given text and also preserving text-irrelevant contents of the original image. To fuse text and image information, existing methods (Dong et al., 2017; Nam et al., 2018) typically choose to directly concatenate image and global sentence features along the channel direction. Albeit simple, the above heuristic may suffer from some potential issues. Firstly, the model cannot precisely correlate fine-grained words with corresponding visual attributes that need to be modified, leading to inaccurate and coarse modification. For instance, shown in the first row of Fig. 4.9, baselines cannot generate detailed visual attributes like *black eye rings* and a *black bill*. Secondly, the model cannot effectively identify text-irrelevant contents and thus fails to reconstruct them, resulting in undesirable modification of text-irrelevant parts in the image. For example, in Fig. 4.9, besides modifying the required attributes, baselines also change the texture of the bird (first row) and the structure of the scene (second row).

Additionally, the attention-based connection introduced in Section 4.1 can only establish correlations between visual and textual information, but it lacks the ability to focus on text-irrelevant image contents. In other words, it does not effectively capture image regional information that is not mentioned in the given text.

To address the above issues, it is important to correctly correlate image visual attributes with corresponding semantic words from a given text description, which allows the network to identify specific image regions, and then achieve an effective modification, aligned with the given text description. In the following, we give more details about our proposed text-image affine combination module, based on an affine transformation, to achieve an effective interaction.

4.2.2 Affine Transformation Based Text-Guided Image Manipulation

To achieve an effective text-guided image manipulation, we propose a novel generative adversarial network for text-guided image manipulation (ManiGAN), which can generate high-quality new attributes matching the given text, and at the same time effectively reconstruct text-irrelevant contents of the original image. In the following, we present more details about key components used in our proposed network.

4.2.2.1 Architecture

As shown in Fig. 4.10, we adopt the multi-stage ControlGAN architecture as the basic framework (see Section 4.1), as it achieves high-quality and controllable image

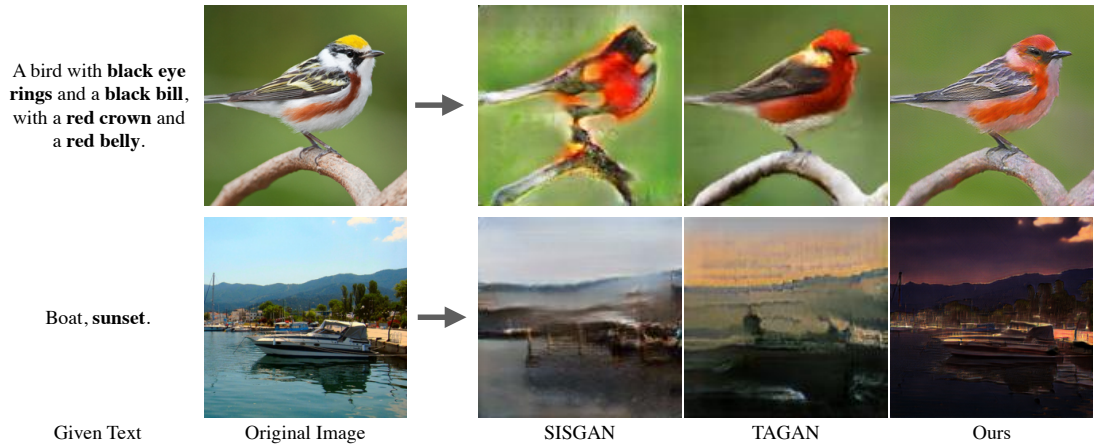


Figure 4.9: Given an image that needs to be edited and a text provided by a user describing desired attributes, the goal is to edit parts of the image according to the given text while preserving text-irrelevant contents. State-of-the-art methods at the time of writing this paper (Dong et al., 2017; Nam et al., 2018) are only capable of generating low-quality images. In these images, the modified objects appear distorted and blurry, and text-irrelevant contents, such as the background, are also modified. In contrast, our method achieves accurate manipulation of the image to align with the provided description, while also effectively reconstructing the text-irrelevant contents.

generation results based on the given text descriptions. We add an image encoder, which is a pre-trained Inception-v3 network (Szegedy et al., 2016), to extract regional image representations v. Our proposed text-image affine combination module (ACM) is utilized to fuse text representations, encoded from a pre-trained RNN (Nam et al., 2018), and regional image representations before each upsampling block at the end of each stage. For each stage, the text features are refined with several convolutional layers and also spatial attention and channel-wise attention introduced in Section 4.1 to produce hidden features h. The proposed ACM further combines h with the original image features v in order to effectively select image regions corresponding to the given text, and then correlate those regions with text information for accurate manipulation. Meanwhile, it also encodes the original image representations for stable reconstruction. The output features from the ACM module are fed into the corresponding generator to produce an edited image, and are also upsampled serving as input to the next stage for image manipulation at a higher resolution. The whole framework gradually generates new visual attributes matching the given text description at a higher resolution with higher quality, and also reconstructs text-irrelevant contents existing in the input image at a finer scale. Finally, the proposed detail correction module (DCM) is used to rectify inappropriate attributes, and to complete missing details.

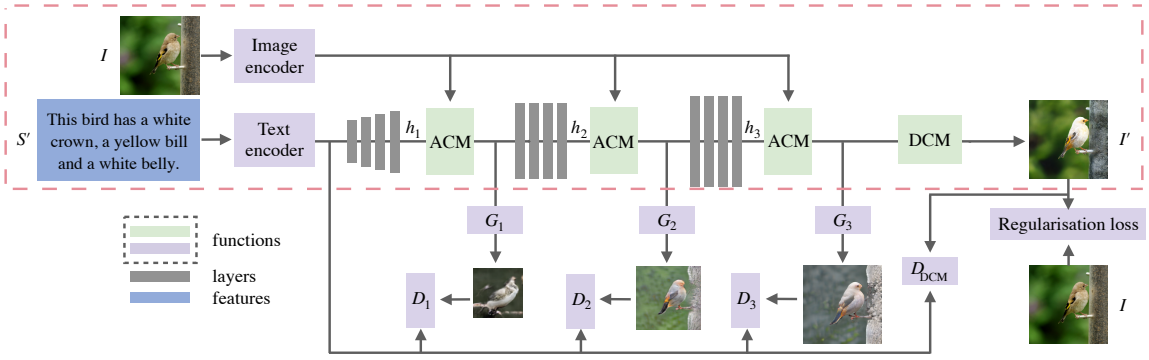


Figure 4.10: Architecture of ManiGAN. It has three stages, and each stage has a generator and a discriminator, which progressively generates images from coarse to fine-grained scales. At each stage, we use proposed ACM to fuse both visual and textual information. The red dashed box indicates the inference pipeline, where a user input both an image and a text description S' that contains desired image attributes, and then the pipeline generates required modified images progressively. However, in training, the text S' is replaced by S that correctly describes I. ACM denotes the text-image affine combination module. DCM denotes the detail correction module. The attention is omitted for simplicity.

4.2.2.2 Text-Image Affine Combination Module

The concatenation-based method used in previous works (Dong et al., 2017; Nam et al., 2018) to combine text and image representations from different modalities is not effective in accurately identifying the regions of the image that need to be modified. As a result, these methods struggle to achieve fine-grained image manipulation, both in terms of generating high-quality attributes corresponding to the given text and maintaining the stability of text-irrelevant image contents during reconstruction. Similarly, the attention-based connection described in Section 4.1 also overlooks text-irrelevant contents. To address these limitations, we propose a simple text-image affine combination module to fuse text-image cross-modality representations, where text and image features collaborate to select text-relevant regions that need to be modified, and then correlate those regions with corresponding semantic words for generating new visual attributes semantically aligned with the given text description. Meanwhile, it also encodes original image for reconstructing text-irrelevant contents.

As shown in Fig. 4.11 (a), our affine combination module takes two inputs: (1) the hidden features $h \in \mathbb{R}^{C \times H \times D}$ from the input text or intermediate hidden representations between two stages, where C is the number of channels, H and D are the height and width of the feature map, respectively, and (2) the regional image features $v \in \mathbb{R}^{256 \times 17 \times 17}$ encoded from the input image I by the Inception-v3 (Szegedy et al.,

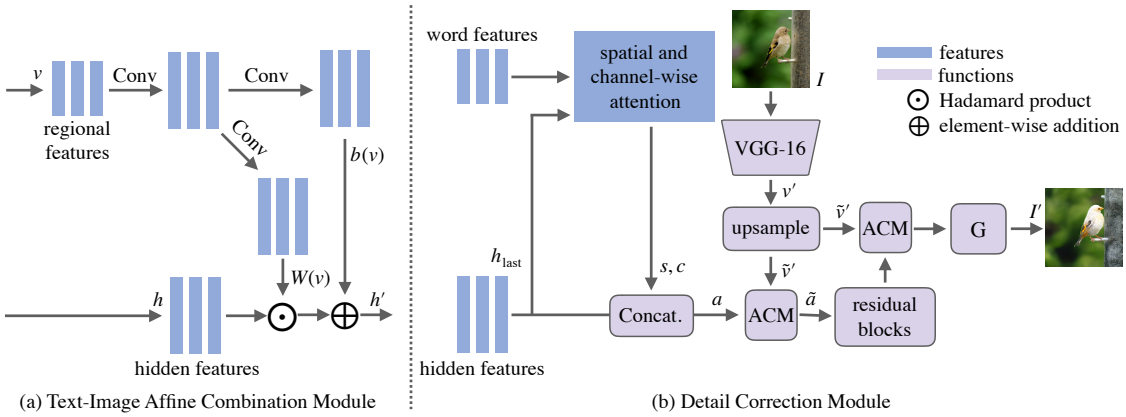


Figure 4.11: Architecture of the text-image affine combination module and the detail correction module. In (b), ACM denotes the text-image affine combination module.

2016). Then, v is upsampled and further processed with two convolutional layers to produce $W(v)$ and $b(v)$ that have the same size as h. Finally, we fuse the two modality representations to produce $h' \in \mathbb{R}^{C\times H\times D}$ as

$$h' = h \odot W(v) + b(v), \tag{4.9}$$

where $W(v)$ and $b(v)$ are the learned weights and biases based on the regional image features v, which are used in the proposed ACM to update values of hidden features, and $\odot$ denotes Hadamard element-wise product. We use W and b to represent the functions that convert the regional features v to scaling and bias values.

Our affine combination module (ACM) is designed to learn to fuse text and image cross-modality representations. In the module, $W(v)$ and $b(v)$ are used to encode the input image into semantically meaningful features, as illustrated in Fig. 4.12. Through experimentation, we have observed that the multiplication operation allows the image feature maps extracted from the input image to re-weight the hidden features h generated from the given text. This serves the purpose of regional selection, enabling the model to precisely identify the desired attributes (such as parts of a bird) that align with the given text. At the same time, it establishes a correlation between attributes and semantic words, facilitating effective manipulation. The bias term in the module encodes image information, ensuring the stable reconstruction of text-irrelevant contents (as shown in Fig. 4.12, $b(v)$, where only text-irrelevant contents, like the background, are highlighted). The above is in contrast with previous approaches (De Vries et al., 2017; Dumoulin et al., 2016; Huang and Belongie, 2017; Park et al., 2019) which apply conditional affine transformation in normalization layers to compensate potential information loss due to normalization (Park et al., 2019) or

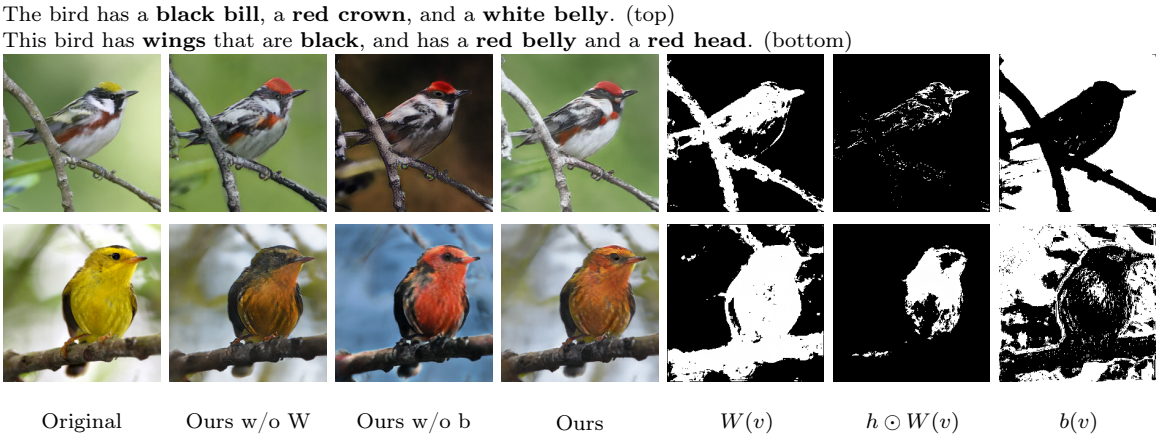


Figure 4.12: Ablation studies of the learned W and b. The texts on the top are the given descriptions containing desired visual attributes, and the last three columns are the channel feature maps of $W(v)$, $h \odot W(v)$, and $b(v)$.

to incorporate style information from a style image (Dumoulin et al., 2016; Huang and Belongie, 2017). To better understand what has been actually learned by different components of our proposed affine combination module, we give a deeper analysis shown in Section 4.2.3.2.

Why does the affine combination module work better than concatenation? By simply concatenating the text and image representations along the channel direction, existing models cannot explicitly differentiate regions that need to be modified from those that need to be preserved, making it challenging to strike a balance between generating new attributes and reconstructing original content. Consequently, this imbalance often leads to inaccurate or coarse modifications, as well as unintended changes to text-irrelevant content. In contrast, our affine combination module employs multiplication between text and image representations, enabling regional selection and assisting the model in focusing on generating the desired fine-grained visual attributes. Additionally, the additive bias component encodes text-irrelevant image information, facilitating the reconstruction of content that does not require modification.

4.2.2.3 Detail Correction Module

To further enhance the details and complete missing contents in the synthetic image, we propose a detail correction module (DCM), exploiting word-level text information and fine-grained image features.

As shown in Fig. 4.11 (b), our detail correction module takes three inputs: (1) the last hidden features $h_{\text{last}} \in \mathbb{R}^{C' \times H' \times D'}$ from the last affine combination module, (2) the word features encoded by a pre-trained RNN following Xu et al. (2018), where each

word is associated with a feature vector, and (3) visual features $v' \in \mathbb{R}^{128\times128\times128}$ that are extracted from the input image I, which are the *relu2_2* layer representations from a pre-trained VGG-16 (Simonyan and Zisserman, 2014) network.

Firstly, to further incorporate fine-grained word-level representations into hidden features h_{last}, we adopt the spatial attention and channel-wise attention introduced in Section 4.1 to generate spatial and channel-wise attention features $s \in \mathbb{R}^{C'\times H'\times D'}$ and $c \in \mathbb{R}^{C'\times H'\times D'}$, respectively, which are further concatenated with h_{last} to produce intermediate features a. The features a can further aid the model to refine visual attributes that are relevant to the given text, contributing to a more accurate and effective modification of the contents corresponding to the given description. Secondly, to introduce detailed visual features from the input image for high-quality reconstruction, the shallow representations v' of layer *relu2_2* from the pre-trained VGG network are utilized, which are further upsampled to be the same size as a, denoted as $\tilde{v}'$. Then, our proposed affine attention module is utilized to fuse visual representations $\tilde{v}'$ and hidden representations a, producing features $\tilde{a}$. Finally, we refine $\tilde{a}$ with two residual blocks to generate the final manipulated image I'.

Why does the detail correction module work? This module aims to refine the manipulated results by enhancing details and completing missing contents. On the one hand, the word-level spatial and channel-wise attentions closely correlate fine-grained word-level information with the intermediate feature maps, enhancing the detailed attribute modification. On the other hand, the shallow neural network layer is adopted to derive visual representations, which contain more detailed color, texture, and edge information, contributing to missing detail construction. Finally, further benefiting from our ACM, the above fine-grained text-image representations collaborate to enhance the quality.

4.2.2.4 Training

First, our network has a multi-stage architecture that enables it to learn the fundamental features of an image at a low resolution and then refine those features at higher resolutions, resulting in a more detailed and realistic final image. Then, to train the network, we use adversarial training, where we alternately optimize our multi-stage network and the discriminators (D_1, D_2, D_3, D_{DCM}).

Generator objective. We adopt the generator objective introduced in Section 4.1 for training the generator. Besides, we add a regularization term as

$$\mathcal{L}_{\text{reg}} = 1 - \frac{1}{CHW}||I' - I||, \tag{4.10}$$

where I is the real image sampled from the true image distribution, and I' is the corresponding modified result produced by our model. The regularization term can produce a large penalty when the generated image I' is the same as the input image, which ensures diversity and prevents the network learning an identity mapping.

Discriminator objective. The objective is same as the objective introduced in Section 4.1, and the function used to train the discriminator in the detail correction module is the same as the one used in the last stage of the main module.

Training. Differently from loss objectives introduced in Section 4.1, which has paired sentence S and corresponding ground-truth image I provided by CUB and COCO datasets for training text-guided image generation models to learn the mapping $S \rightarrow I$, both datasets do not provide paired training data $(I, S') \rightarrow I'_{\text{gt}}$ for training text-guided image manipulation models, where S' is a text describing new attributes, and I'_{gt} is the corresponding ground truth modified image.

To simulate the training data, we use paired data $(I, S) \rightarrow I$ to train the model. A natural question may arise: how does the model learn to modify the image I if the input image I and ground-truth image are the same, and the modified sentence S' does not exist in the input? In theory, the optimal solution is that the network becomes an identity mapping from the input image to the output. The text-guided image manipulation model is required to jointly solve image generation from text descriptions $(S \rightarrow I)$, and text-irrelevant contents reconstruction $(I \rightarrow I)$. Thanks to our proposed affine combination module, our model gains the capacity to disentangle regions required to be edited and regions needed to be preserved. Also, to generate new contents semantically matching the given text, the paired data S and I can serve as explicit supervision.

Moreover, to prevent the model from learning an identity mapping and to promote the model to learn a good $(S \rightarrow I)$ mapping in the regions relevant to the given text, we propose the following training schemes. Firstly, we introduce a regularization term $\mathcal{L}_{\text{reg}}$ as Eq. (4.10) in the generator objective to produce a penalty if the generated image becomes the same as the input image. Secondly, we choose to early stop the training when the model achieves the best trade-off between the generation of new visual attributes aligned with the given text descriptions and the reconstruction of text-irrelevant contents existing in the original images. The stop criterion is determined by evaluating the model on a hold-out validation and measuring the results by our proposed image manipulation evaluation metric, called manipulative precision (see Fig. 4.13), which is discussed in Section 4.2.3.

4.2.3 Experiments

Table 4.2: Quantitative comparison: Inception score (IS), text-image similarity (sim), L_1 pixel difference (diff), and manipulative precision (MP) of state-of-the-art approaches and ManiGAN on the CUB and COCO. "w/o ACM" denotes without the affine combination module. "w/ Concat." denotes using concatenation method to combine hidden and image features. "w/o main" denotes without main module. "w/o DCM" denotes without detail correction module. For IS, similarity, and MP, higher is better; for pixel difference, lower is better.

	CUB				COCO			
Method	IS	sim	diff	MP	IS	sim	diff	MP
SISGAN	2.24	.045	.508	.022	3.44	.077	.442	.042
TAGAN	3.32	.048	.267	.035	3.28	.089	.545	.040
Ours w/o ACM	4.01	**.138**	.491	.070	5.26	.121	.537	.056
Ours w/ Concat.	3.81	.135	.512	.065	13.48	.085	.532	.039
Ours w/o main	**8.48**	.084	**.235**	.064	**17.59**	.080	**.169**	.066
Ours w/o DCM	3.84	.123	.447	.068	6.99	**.138**	.517	.066
Ours	8.47	.101	.281	**.072**	14.96	.087	.216	**.068**

Our model is evaluated on the CUB bird (Wah et al., 2011) and more complicated COCO (Lin et al., 2014) datasets, comparing with two state-of-the-art approaches SISGAN (Dong et al., 2017) and TAGAN (Nam et al., 2018) on image manipulation using natural language descriptions.

Implementation. In our setting, we train the detail correction module (DCM) separately from the main module. Once the main module has converged, we train the DCM subsequently and set the main module as the eval mode. There are three stages in the main module, and each stage contains a generator and a discriminator. We train three stages at the same time, and three different-scale images $64 \times 64, 128 \times 128, 256 \times 256$ are generated progressively.

The main module is trained for 600 epochs on CUB and 120 epochs on COCO using the Adam optimizer (Kingma and Ba, 2014) with the learning rate 0.0002, and $\beta_1 = 0.5$, $\beta_2 = 0.999$. As for the detail correction module, there is a trade-off between the generation of new attributes corresponding to the given text and the reconstruction of text-irrelevant contents of the original image. Based on the manipulative precision (MP) values (see Fig. 4.13), we find that training 100 epochs for CUB, and 12 epochs for COCO to achieve an appropriate balance between generation and reconstruction. The other training setting is the same as in the main module. The hyperparameter controlling $\mathcal{L}_{\text{reg}}$ in Eq. (4.10) is set to 1 for CUB and 15 for COCO.

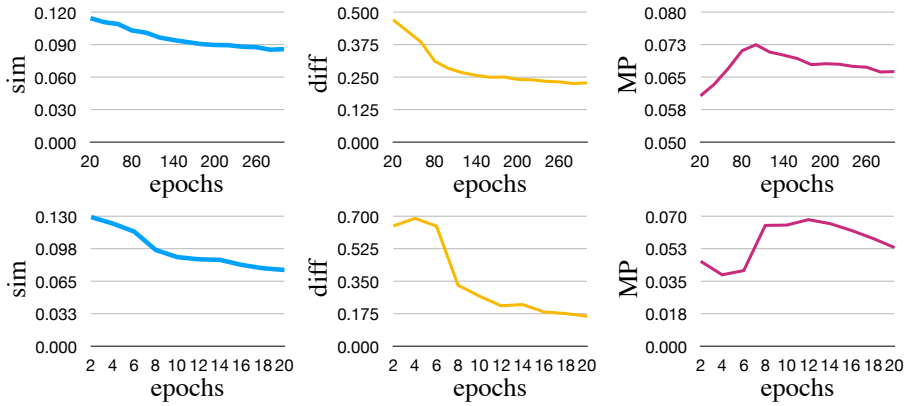


Figure 4.13: Text-image similarity (sim), L_1 pixel difference (diff), and manipulative precision (MP) values at different epochs on the CUB (top) and COCO (bottom) datasets. We suggest to stop training the DCM module when the model gets the highest MP values shown in the last column.

Manipulative precision metric. Image manipulation using natural language descriptions should be evaluated in terms of both the generation of new visual attributes from the given text, and the reconstruction of original contents existing in the input image. However, existing metrics only focus on one aspect of this problem. For example, the L_1 Euclidean distance, Peak Signal-to-Noise Ratio (PSNR), and SSIM (Wang et al., 2004) only measure the similarity between two images, while the cosine similarity and the retrieval accuracy (Li et al., 2019a; Nam et al., 2018; Xu et al., 2018) only evaluate the similarity between the text and the corresponding generated image. Based on this, we contribute a new metric, called manipulative precision (MP), for this area to simultaneously measure the quality of generation and reconstruction. The metric is defined as

$$\text{MP} = (1 - \text{diff}) \times \text{sim}, \tag{4.11}$$

where diff is the L_1 pixel difference between the input image and the corresponding modified image, sim is the text-image similarity, which is calculated by using pre-trained text and image encoders (Xu et al., 2018) based on a text-image matching score to extract global feature vectors of a given text description and the corresponding modified image, and then the similarity value is computed by applying cosine similarity between these two global vectors. Specifically, the design is based on the intuition that if the manipulated image is generated from an identity mapping network, then the text-image similarity should be low, as the synthetic image cannot perfectly keep a semantic consistency with the given text description.

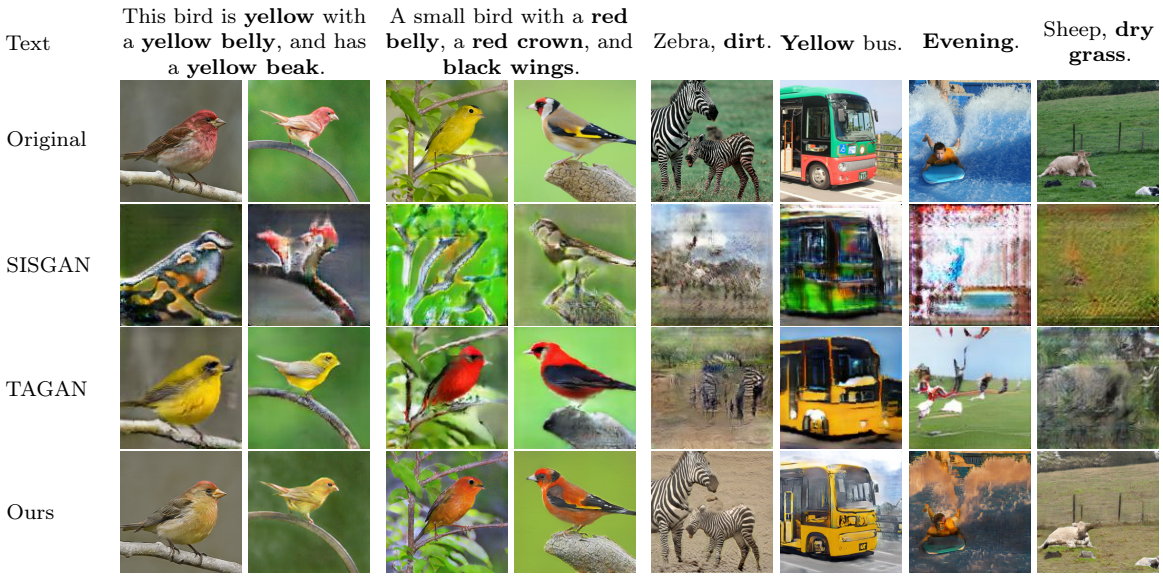


Figure 4.14: Qualitative comparison of three methods on the CUB and COCO.

4.2.3.1 Comparison with State-of-the-Art Approaches

Quantitative comparison. As mentioned above, our model can generate high-quality images compared with the state-of-the-art methods. To demonstrate this, we adopt the Inception score (IS) (Salimans et al., 2016) as a quantitative evaluation measure. Besides, we further adopt manipulative precision (MP) to evaluate manipulation results. In our experiments, we evaluate the IS on a large number of manipulated samples generated from mismatched pairs, i.e., randomly chosen input images manipulated by randomly selected text descriptions.

As shown in Table 4.2, our method has the highest IS and MP values on both the CUB and COCO datasets compared with the state-of-the-art approaches, which demonstrates that (1) our method can produce high-quality manipulated results, and (2) our method can better generate new attributes matching the given text, and also effectively reconstruct text-irrelevant contents of the original image.

Qualitative comparison. Fig. 4.14 shows the visual comparison between our ManiGAN, SISGAN (Dong et al., 2017), and TAGAN (Nam et al., 2018) on the CUB and COCO datasets. It can be seen that both state-of-the-art methods are only able to produce low-quality results and cannot effectively manipulate input images on the COCO dataset. However, our method is capable of performing an accurate manipulation and also keep a highly semantic consistency between synthetic images and given text descriptions, while preserving text-irrelevant contents. For example, shown in the last column of Fig. 4.14, SISGAN and TAGAN both fail to achieve an



Figure 4.15: Ablation studies. a: given text describing the desired visual attributes; b: input image; c: removing all ACMs and DCM, only concatenating image and text features before feeding into the main module; d: using the concatenation method to replace all ACMs; e: removing the main module and just training DCM only; f: removing DCM and just training the main module only; g: our full model.

effective manipulation, while our model modifies the *green grass* to *dry grass* and also edits the *cow* into a *sheep*.

Note that as birds have detailed descriptions (e.g., color for different parts), we use a long sentence to manipulate them, while the descriptions for COCO are more abstract and focus on categories, thus we use words (i.e., object + desired attributes) to do manipulation for simplicity, which has the same effect as using a sentence.

4.2.3.2 Ablation Studies

Ablation experiments of the affine combination module. To better understand what has been learned by our ACM, we ablate and visualize the learned feature maps shown in Fig. 4.12. As we can see, without W, some attributes cannot be perfectly generated (e.g., white belly in the first row and red head in the second row), and without b, the text-irrelevant contents (e.g., background) are hard to preserve, which verifies our assumption that W behaves as a regional selection function to help the model focus on attributes corresponding to the given text, and b helps to complete

missing text-irrelevant details of the original image. Also, the visualization of the channel feature maps of $W(v)$, $h \odot W(v)$, and $b(v)$ shown in the last three columns of Fig. 4.12 validates the regional selection effect of the multiplication operation.

Effectiveness of the affine combination module. To verify the effectiveness of the ACM, we use the concatenation method to replace all ACMs, which concatenates hidden features h and regional features v along the channel direction, shown in Fig. 4.15 (d). As we can see, with the concatenation method, the model generates structurally different birds on CUB, and fails to do manipulation on COCO, which indicates that it is hard for the concatenation method to achieve a good balance between generation and reconstruction. The results on CUB is an example of the generation effect surpassing the reconstruction effect, while results on COCO show the domination of the reconstruction effect. In contrast, due to the regional selection effect of ACM that can distinguish which parts need to be generated or to be reconstructed, our full model synthesizes an object having the same shape, pose, and position as the one existing in the original image, and also generates new visual attributes aligned with the given text description.

Also, to further validate the effectiveness of ACM, we conduct an ablation study shown in Fig. 4.15 (c). In "Our w/o ACM", we fully remove ACM in the main module and remove DCM as well. That is the main module without ACM, and we only concatenate original image features with text features at the beginning of the model and do not further provide additional original image features in the middle of the model. This method is used in both state-of-the-art SISGAN (Dong et al., 2017) and TAGAN (Nam et al., 2018). It can be seen that our model without ACM fails to produce realistic images on both datasets. In contrast, our full model better generates attributes matching the given text, and also reconstructs text-irrelevant contents shown in (g). Table 4.2 also verifies the effectiveness of our ACM, as the values of IS and MP increase significantly when we implement ACM.

Effectiveness of the detail correction module and main module. As shown in Fig. 4.15 (f), our model without DCM misses some attributes (e.g., the bird missing the tail in the second row, the zebra missing the mouth in the third row), or generates new contents (e.g., new background in the first row, different appearance of the bus in the fourth row), which indicates that our DCM can correct inappropriate attributes and reconstruct text-irrelevant contents. Fig. 4.15 (e) shows that without main module, our model fails to do image manipulation on both datasets, which just achieves an identity mapping. This is mainly because the model fails to correlate words with corresponding attributes, which has been done in the main module. Table 4.2 also

illustrates the identity mapping, as our model without main module gets the lowest L_1 pixel difference value.

4.2.4 Conclusion

We have proposed a novel generative adversarial network for image manipulation, called ManiGAN, which can semantically manipulate input images using natural language descriptions. Two novel components are proposed: (1) the affine combination module selects image regions according to the given text, and then correlates the regions with corresponding semantic words for effective manipulation. Meanwhile, it encodes original image features for text-irrelevant contents reconstruction. (2) The detail correction module rectifies mismatched visual attributes and completes missing contents in the synthetic image. Extensive experimental results demonstrate the superiority of our method, in terms of both the effectiveness of image manipulation and the capability of generating high-quality results.

4.3 Memory-Driven Approach

In the previous two sections, we introduced two different approaches for text-to-image generation. First, we presented an attention-based connection (as described in Section 4.1), which allows the network to disentangle different visual attributes and provides control over the generation of specific parts of the image. Then, we introduced an affine-transformation-based connection (as discussed in Section 4.2) for text-guided image manipulation, which effectively combines text and image information while preserving text-irrelevant image details.

Building upon the success of fusing real images into the generation pipeline in text-guided image manipulation (as shown in Section 4.2), we propose to also incorporate real images into the generation pipeline for text-to-image generation. This approach leverages the benefits of utilizing real image features to enhance the generation process.

A question arises: "How can we incorporate real images during inference without requiring users to provide an image in the text-to-image generation task?" To address this challenge, we propose a semi-parametric text-image correlation approach. This approach involves constructing a memory bank to store candidate image features. When given a text input, we retrieve the corresponding image features from the memory bank and integrate them into the generation pipeline.

In the following sections, we provide an overview of the semi-parametric method in the text-to-image generation task and introduce the basic background. We then delve into the details of our proposed semi-parametric memory-driven approach.

4.3.1 Background of Semi-Parametric Method in Text-to-Image Generation

At the time of writing this paper, significant progress has been made in text-to-image generation with the development of generative adversarial networks (GANs) (Goodfellow et al., 2014; Gauthier, 2015; Mirza and Osindero, 2014). These parametric models, which utilize deep networks and their weights to represent realistic visual appearances, have achieved impressive results on well-structured datasets that focus on specific object classes, such as birds (Wah et al., 2011) and flowers (Nilsback and Zisserman, 2008), when accompanied by detailed descriptions. However, there is still ample room for improvement, as these approaches often struggle when applied to more complex datasets containing diverse backgrounds and multiple objects, such as COCO (Lin et al., 2014). The wide variety of objects, backgrounds, and scene compositions in COCO poses a challenge for these approaches, making it difficult to

capture intricate geometric structures and generate appearances that are consistent with the given text semantics. As illustrated in Fig. 4.16, the baseline models can only produce appearances that match the given semantics (e.g., black and white stripes for a zebra), but the generated objects appear unrealistic and exhibit distorted shapes.

Moreover, these parametric approaches differ from earlier works in image synthesis that relied on *non-parametric* techniques, which could leverage large datasets of images during the inference stage (Chen et al., 2009; Hays and Efros, 2007; Isola and Liu, 2013; Zhu et al., 2015; Lalonde et al., 2007). While parametric approaches offer the advantages of end-to-end training and highly expressive models, they may overlook the benefits of earlier non-parametric techniques, as they fail to harness the vast amount of image data available at inference time. However, by incorporating real images during inference, we can facilitate the generation process by providing valuable information about the target images, such as their semantic characteristics (e.g., the black and white appearance of a zebra or the rectangular shape of a bus).

In this section, we introduce a semi-parametric memory-driven approach for text-to-image generation. This approach combines the strengths of both parametric and non-parametric techniques. The non-parametric component consists of a memory bank that stores image features extracted from a training set of real images. The parametric component is a generative adversarial network. At inference, when given a description, the memory bank is used to selectively retrieve relevant image features. These features serve as foundational information, allowing the generator to directly incorporate cues about the target images and produce realistic synthetic results. Further discussion on this topic can be found in Section 4.3.2.2.

4.3.2 Memory-Driven Generative Adversarial Networks

In the following, we first show the overview of our entire network, and then give more details about the proposed semi-parametric memory-driven approach, with a new discriminator using content information.

4.3.2.1 Overview

Given a sentence S, our objective is to generate a synthetic image I' that is semantically aligned with S. To achieve this, our semi-parametric approach is trained on a dataset of paired text descriptions and real image features v, denoted as (S, v). The image features v are extracted from real images using a pre-trained VGG-16 network (Simonyan and Zisserman, 2014). These paired sets of text descriptions and image features (S, v)

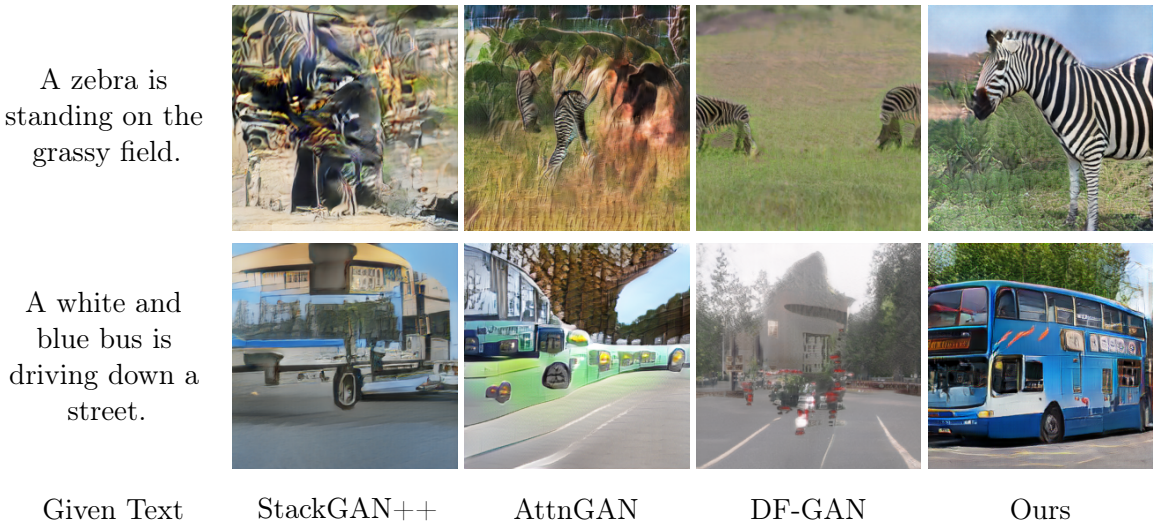


Figure 4.16: Examples of text-to-image generation on the COCO. Current approaches only generate low-quality images with unrealistic objects. In contrast, our method can produce realistic images, in terms of both visual appearances and geometric structure.

are used to construct a memory bank M. Each element in M represents an image feature extracted from a training image and is associated with the corresponding semantically-matched text descriptions from the training dataset.

At inference time, we are provided with a new text description S that was not seen during training. We use S to retrieve semantically-aligned image features from the memory bank M using matching algorithms. These retrieved image features v and word embeddings w, which are encoded from the given text description S, are then fed into the generator to synthesize an output image.

During the generation process, the generator has the ability to selectively choose relevant semantic information from the retrieved image features v and fuse them with hidden features to produce realistic images that are semantically aligned with S. This allows the generator to incorporate specific visual details and attributes based on the provided text description, resulting in more accurate and meaningful image synthesis.

The complete architecture is depicted in Fig. 4.17. In this architecture, we employ a multi-stage approach (as described in Section 2.1.5), where the network progressively generates images from coarse to fine scales.

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 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

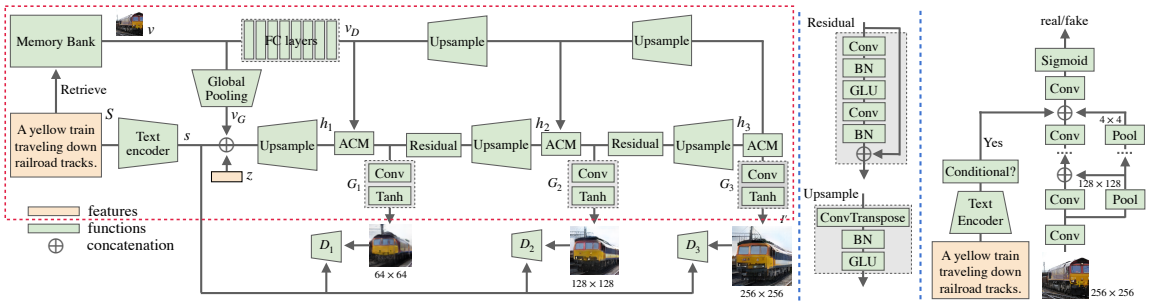


Figure 4.17: Left: architecture of our proposed method. The red box indicates the inference pipeline that retrieves image features from a memory bank according to the given description S; during training, we directly feed image features from the text-paired training image. z is a random vector drawn from the Gaussian distribution. Right: architecture of the proposed discriminator with content information.

unseen text to better retrieve image features at inference time.

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.

4.3.2.2 Generator with Image Features

To incorporate image features into the generation pipeline, we start by applying average pooling to v along each channel, which filters potential content details (such as overall spatial structure) contained in v. This process yields a global image feature v_G that retains only the basic information of the corresponding real image I, serving as the foundational image priors. By including v_G in the generation process, the model can effectively avoid simply copying and pasting from I, ensuring the diversity of the generated output. This is particularly crucial in the first stage of the multi-stage architecture, as subsequent stages focus on refining the initial images by adding more details and improving their resolution. For instance, as illustrated in Fig. 4.17, the synthetic image generated by the first stage (G_1) establishes the fundamental structure of the target image, encompassing attributes such as color, pose, and size of a train.

Subsequent stages primarily concentrate on enhancing the quality and fine-grained details of the train in the generated image.

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) (Li et al., 2020a), 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? The generator's goal is to generate samples, such as images, that are indistinguishable from the training distribution, which is based on real samples from the training dataset. By incorporating image features from real images into the generator, it can draw cues and information from the desired distribution it needs to generate. Furthermore, the global feature v_G and image features v_D provide preliminary information about the target results and act as candidate information, enabling the model to selectively choose text-required information without having to generate it from scratch. This simplifies the generation process and enhances the model's ability to produce realistic and accurate outputs. To some extent, the global feature v_G can be considered as the metadata of the target image, which may contain information about the type of object to generate, such as zebra or bus. On the other hand, v_D provides basic object information, such as preserving the spatial structure of the zebra, including its four legs and one head, in the generated images, as well as the rectangle shape for the bus, by leveraging our proposed method.

4.3.2.3 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 (i.e., image features are extracted from shallow layers of a neural network). 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 (Gatys et al., 2016; Johnson et al., 2016). 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. (2014a) 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 $a_{128} \in \mathbb{R}^{C\times128\times128}$ to $a_4 \in \mathbb{R}^{C\times4\times4}$, 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.

4.3.2.4 Training

The generator and discriminator are trained alternatively by minimizing the generator loss $\mathcal{L}_G$ and discriminator loss $\mathcal{L}_D$.

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 $\mathcal{L}_{\text{DAMSM}}$ (Xu et al., 2018), where unconditional loss checks the image quality of output results, conditional loss checks the semantic alignment between output images and the given texts, and $\mathcal{L}_{\text{DAMSM}}$ is used to check whether word-related image attributes are produced in synthetic images.

$$\begin{aligned}\mathcal{L}_{G_i} = &\underbrace{-\frac{1}{2}E_{z\sim P_z, v\sim P_{\text{data}}}\left[\log(D_i(G_i(z,S,v)))\right]}_{\text{unconditional adversarial loss}} \\ &\underbrace{-\frac{1}{2}E_{z\sim P_z, v\sim P_{\text{data}}}\left[\log(D_i(G_i(z,S,v),S))\right]}_{\text{conditional adversarial loss}} + \lambda\mathcal{L}_{\text{DAMSM}},\end{aligned} \tag{4.12}$$

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:

$$\mathcal{L}_G = \sum_{k=1}^{K} (\mathcal{L}_{G_i}), \tag{4.13}$$

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

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

$$\begin{aligned} \mathcal{L}_{D_i} = & \underbrace{-\frac{1}{2} E_{I_i \sim P_{\text{data}}} \left[\log(D_i(I_i))\right] - \frac{1}{2} E_{z \sim P_z} \left[\log(1 - D_i(G_i(z, S, v)))\right]}_{\text{unconditional adversarial loss}} \\ & \underbrace{-\frac{1}{2} E_{I_i \sim P_{\text{data}}} \left[\log(D_i(I_i, S))\right] - \frac{1}{2} E_{z \sim P_z} \left[\log(1 - D_i(G_i(z, S, v), S))\right]}_{\text{conditional adversarial loss}}, \end{aligned} \tag{4.14}$$

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

$$\mathcal{L}_D = \sum_{k=1}^{K} (\mathcal{L}_{D_i}) + R_1(\psi), \tag{4.15}$$

where $R_1(\psi)$ is a regularization term described in the paper (Mescheder et al., 2018). This regularization term is derived from zero-centered gradient penalties (Ross and Doshi-Velez, 2017) 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.

4.3.3 Experiments

To verify the effectiveness of our proposed method in realistic image generation from text descriptions, we conducted extensive experiments on the CUB bird (Wah et al., 2011) and the more complex COCO (Lin et al., 2014) dataset, where COCO contains multiple objects with diverse backgrounds.

Table 4.3: Quantitative comparison: IS, FID, R-precision, SOA-C, and SOA-I of current methods and our approach on the CUB and COCO. 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	4.36	23.98	67.82	25.89	32.32	85.47	25.88	39.01
ControlGAN	4.58	13.92	69.33	24.06	33.58	72.43	-	-
MirrorGAN	4.56	-	57.67	26.47	-	74.52	-	-
DM-GAN	4.75	16.09	72.31	32.32	32.64	88.56	33.44	48.03
DF-GAN	5.10	14.81	-	-	21.42	-	-	-
XMC-GAN	-	-	-	30.45	9.33	-	50.94	71.33
LAFITE	**5.97**	**10.48**	-	32.34	**8.12**	-	61.09	74.78
DALL-E	-	-	-	-	∼20	-	-	-
GLIDE	-	-	-	-	12.89	-	-	-
CP-GAN	-	-	-	**52.73**	55.82	**93.59**	**77.02**	**84.55**
Obj-GAN	-	-	-	30.29	36.52	87.84	27.14	41.24
OP-GAN	-	-	-	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

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 adjusted based on the size of the input image. A pooling operation with a small stride (stride = 2) and the same-size pooling kernel is applied to the image features. The pooling operation used in this case is average pooling. 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 with the learning rate 0.0002. We preprocess both datasets according to the method used in Xu et al. (2018). Note that No attention module is implemented in the whole architecture.

4.3.3.1 Comparison with Other Approaches

Quantitative comparison. The results are shown in Table 4.3. 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

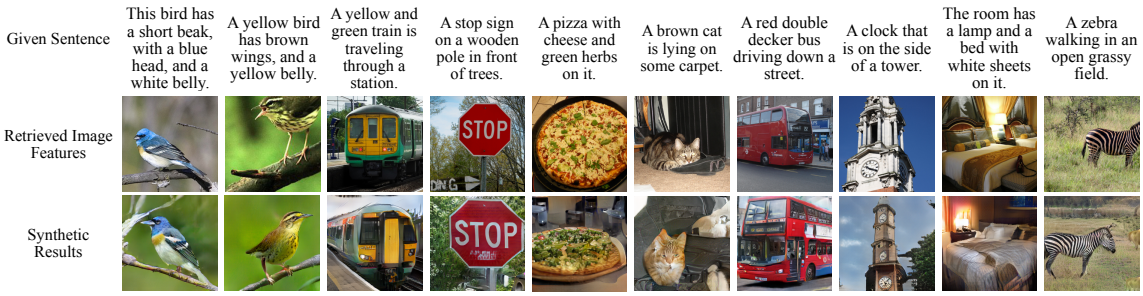


Figure 4.18: Qualitative results on the CUB and COCO: in the top row is the given sentences; middle row: the image features extracted from the memory bank M (we use corresponding images to represent the image features for a better visualization); bottom row: the synthetic results.



Figure 4.19: Qualitative comparison on the COCO.

image quality and diversity, and (2) synthetic results produced by our method are semantically aligned with the given text descriptions. Compared to the large-scale CLIP-based (Radford et al., 2021) 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 (Hinz et al., 2019; Li et al., 2019b; Zhang et al., 2021).

Qualitative comparison. In Fig. 4.18, we present synthetic examples produced



Figure 4.20: Diversity. The top row shows the fixed sentence and the image providing image features. The bottom presents diverse synthetic images produced by solely changing the value of noise vector z.

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. 4.20. When we fix the sentence and image features and only change the value of noise, our method can generate obviously different images with different object location, pose, and background, but they still semantically match the given sentence and also make use of information from the image features, where there are some similarity between the real image and synthetic image.

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. 4.21. 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.

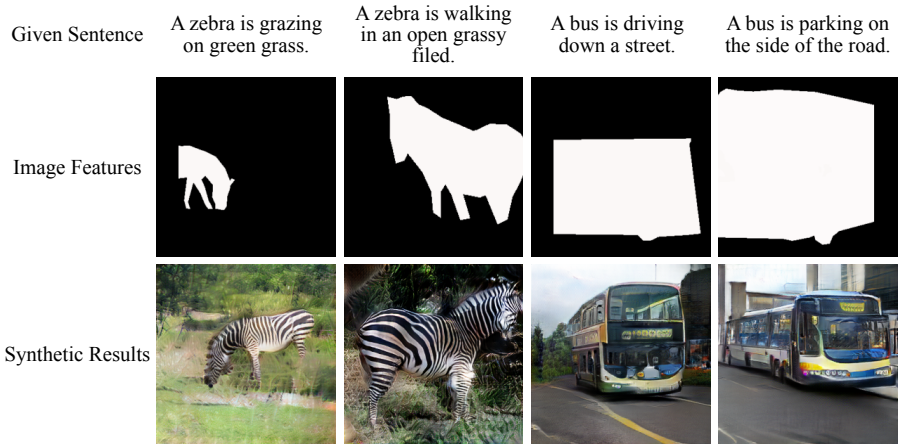


Figure 4.21: Semantic information exploration. Top row: given sentences; middle row: image features, represented by the corresponding segmentation masks for a better visualization; bottom row: synthetic images.

4.3.3.2 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 4.4. 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 incorporating image features, our method becomes a traditional text-to-image generation approach, similar to other baselines. Surprisingly, the FID score of our method without image features, denoted as "Ours w/o Feature" remains competitive with other baselines. This suggests that our method can generate high-quality and diverse synthetic images even without the utilization of image features. We attribute this performance to our improved discriminator, which benefits from the content information and enables more accurate predictions on complex datasets. Consequently, the generator is encouraged to produce better synthetic images.

Effectiveness of the fully connected layers. Here, we show the effectiveness of the fully connected layers (FCLs) applied on the image features v. Interestingly, from Table 4.4, the "model w/o FCLs" 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 FCLs*". As we can see, on mismatched pairs, although FID is still low, the R-precision degrades.

Table 4.4: Ablation studies: "Ours w/o Feature" denotes without feeding image features into the generator, "Ours w/o FCLs" denotes without using the fully connected layers, "Ours w/o FCLs*" 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 FCLs	18.82	92.17
Ours w/o FCLs*	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**

Content information. In Table 4.4, 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 4.4, 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 4.4, 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.

4.4 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 demonstrate the effectiveness of feeding real image features into the generator and incorporating content information into the discriminator.

The proposed semi-parametric approach has introduced an alternative way to enable an effective interaction between multi-modal information, especially achieving an transformation from one domain to the other. Instead of only focusing on directly achieving the cross-domain transformation, our approach shows a promising direction by feeding target real information at the middle of generation pipeline to provide the cues of desired objects and attributes, allowing the generation pipeline to selectively choose required information to eases the whole generation process.

Chapter 5

Lightweight Architecture for Multi-Modal Generative Model

Once a consistent representation of semantically related information is established and effective interaction between modalities is achieved, reducing the implementation cost becomes a crucial challenge. In this chapter, the main focus is on developing a lightweight network for a model that integrates inputs from different modalities. To achieve this, two approaches are proposed: (1) improving the network's capabilities to better capture the correlation between multi-modal information, rather than blindly increasing its size. This involves enhancing the model's ability to extract and integrate relevant features from different modalities, while keeping the network architecture efficient and compact, and (2) redesigning the architecture of different components in the generative model to reduce redundancy. This includes identifying and eliminating unnecessary or redundant components or operations, optimizing the computational efficiency, and streamlining the overall architecture without compromising the model's performance. By employing these approaches, the aim is to strike a balance between model complexity and performance, ultimately creating a lightweight network that efficiently handles multi-modal inputs while achieving high-quality results.

5.1 Improving Capabilities of the Network

In this section, our main focus is on enhancing the capabilities of a generative model, enabling it to have a lightweight architecture while maintaining competitive performance compared to heavier networks. We will begin by introducing the background of lightweight architecture in the context of text-guided image manipulation introduced in Section 2.5.3, before delving into the specific details of how we can improve the capabilities of the network.

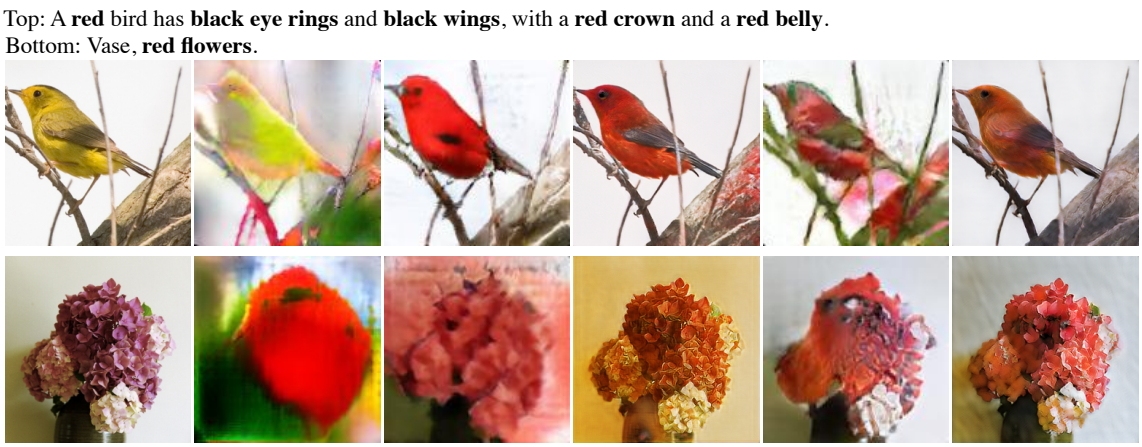


Figure 5.1: Examples of image manipulation using natural language descriptions are shown, demonstrating the effectiveness of our lightweight architecture in accurately matching the input images to the descriptions. The abbreviation "ManiGAN*" indicates that we have reduced the number of stages and parameters in the model.

5.1.1 Background of Lightweight Architecture in Text-Guided Image Manipulation

The goal of this section is to investigate a lightweight architecture of the generative network for text-guided image manipulation, as most current methods require a large memory and need a lot of time for training and inference, which is less practical to memory-limited devices, such as mobile phones. To achieve this, simply reducing the number of stages and parameters in ManiGAN proposed in Section 4.2 cannot achieve a satisfactory result, shown in Fig. 5.1 (e). As we can see, compared with the original ManiGAN (d), the image quality of synthetic results drops significantly as both images are obviously blurred. Also, the manipulation ability of this lightweight ManiGAN becomes worse, because more regions (e.g., sky and branches) are colored red. After a close investigation, we find that the discriminator used in ManiGAN fails to provide the generator with fine-grained training feedback related to each word (i.e., whether the network has successfully generated the image attributes related to all words in the text), because it only calculates the similarity between the whole text and image features. Due to the coarse supervisory feedback from discriminators, the network needs to have a heavy structure with a large number of parameters, to build an accurate relation between visual attributes and corresponding semantic words to achieve an effective manipulation, which greatly impedes the construction of a lightweight architecture. Additionally, even in the ManiGAN, poor training feedback prevents the model from completely disentangling different visual attributes, resulting

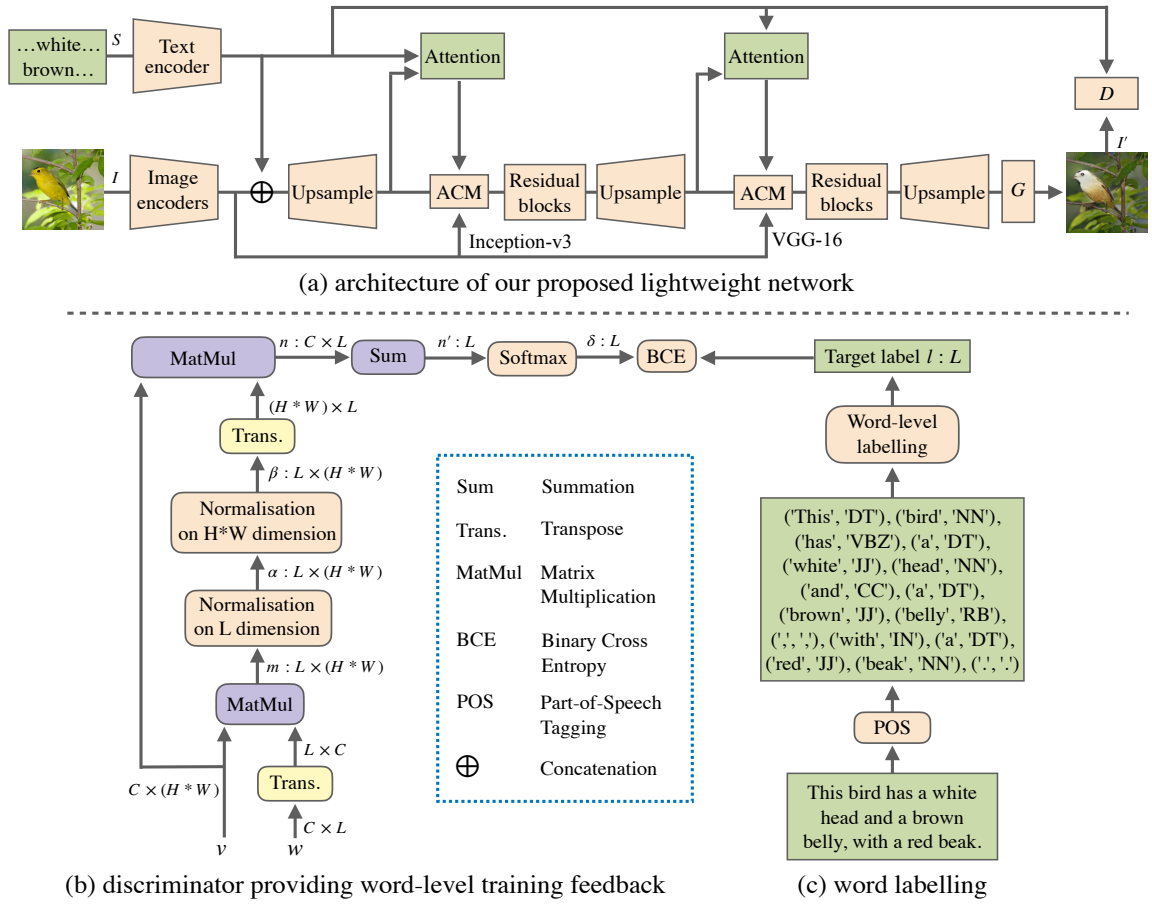


Figure 5.2: Top: architecture of our model. Inception-v3 and VGG-16 denote image features are extracted by the corresponding image encoder. Bottom: the design of our word-level discriminator.

in an incorrect mapping between attributes and their corresponding semantic words. As a result, some text-irrelevant contents may be changed. For example, as shown in Fig. 5.1 (d), the branch is colored red, and the background of the vase is also modified.

Based on this, we propose a new word-level discriminator along with explicit word-level supervisory labels, which can provide the generator with detailed training feedback related to each word, to facilitate training a lightweight generator that has a small number of parameters but can still effectively disentangle different visual attributes, and then correctly map them to the corresponding semantic words. Besides, thanks to our powerful discriminator to provide explicit word-level training feedback, our network can be simplified to have only a generator network and a discriminator network, and we can even further reduce the number of parameters in the whole model by having a lightweight architecture, without sacrificing much image quality, which is more friendly to memory-limited devices.

5.1.2 Lightweight Generative Adversarial Networks for Image Manipulation

To achieve effective text-guided image manipulation while maintaining a lightweight network, we propose a novel word-level discriminator that enables a lightweight architecture. The complete architecture is depicted in Fig. 5.2 (a). In contrast to the architectures proposed in previous sections, the lightweight architecture consists of only a generator and a discriminator. By utilizing a word-level discriminator, we focus on the discriminative power at the word level, allowing for a more efficient and lightweight design. This approach aims to capture the semantic alignment between the generated images and the given text description without the need for additional complex components or heavy computational requirements.

5.1.2.1 Word-Level Discriminator

Although several studies have worked on discriminators at the time of writing this paper, the text-adaptive discriminator (Nam et al., 2018) suffers from the loss of important spatial information in images due to the global average pooling layer. On the other hand, the discriminator proposed in Section 4.1 overlooks word information within text features by directly implementing cosine similarity between the entire text and image features. To address these limitations, we propose a novel word-level discriminator that fully explores the word-level information contained in text features.

5.1.2.2 Word Labelling

To fully explore the word-level information, we label each word in the description using *part-of-speech tagging* (Bird et al., 2009), which marks up each word to a particular part of speech (e.g., noun, verb, adjective, etc.), based on its definition and context. Then, we create a target label l having the same length as the number of words in the sentence, where each **noun** and **adjective** is set to 1 and others to 0. This target label l is used to calculate the loss function (Eq. 5.3) as word-level training feedback. This is achieved by creating an auxiliary classification task that evaluates whether image attributes related to nouns and adjectives are generated in the synthetic images.

Discussions. Given an image, the main observable components are objects and their corresponding visual attributes, which correspond to nouns and adjectives in the description, respectively. The goal is to disentangle the different visual attributes present in the image and establish an accurate relationship between these visual attributes and their corresponding semantic words. The model only needs to identify

the category of objects and their visual attributes and map them to the corresponding semantic words (nouns and adjectives) to establish a correct relationship. Also, there is no need to retain potential logical relationships (such as relative position) contained in the pronoun or verb, as this information is already present in the input image.

5.1.2.3 Discriminator Providing Word-Level Training Feedback

Our novel word-level discriminator takes two inputs: (1) the image features v and v' encoded by the pre-trained Inception-v3 (Szegedy et al., 2016) image encoder from the input and modified images I and I', respectively, and (2) the word features $w \in \mathbb{R}^{C \times L}$ encoded by a pre-trained text encoder introduced in Section 4.1 from the given text description S, where C is the feature dimension, and L is the number of words in S.

Following Section 4.1, for simplicity, we use $v \in \mathbb{R}^{C \times (H*W)}$ to represent both real image features and synthetic image features. First, we compute the word-region correlation matrix $m \in \mathbb{R}^{L \times (H*W)}$ via $m = w^T v$, which contains the information for all pairs of words in the description S and regions of the image I (or I'). Then, we normalize the word-region correlation matrix on both L and $H * W$ dimensions sequentially using the following equations:

$$\alpha_{i,j} = \frac{\exp(m_{i,j})}{\sum_{k=0}^{L-1} \exp(m_{k,j})}, \quad \beta_{i,j} = \frac{\exp(\alpha_{i,j})}{\sum_{k=0}^{(H*W)-1} \exp(\alpha_{i,k})}, \tag{5.1}$$

where $\beta_{i,j}$ represents the probability value that the i^{th} word is relevant to the j^{th} region of the image. Then, the word-weighted image features $n \in \mathbb{R}^{C \times L}$ can be derived via $n = v\beta^T$, where $n_{i,j}$ represents the sum of image features at the i^{th} channel, and each spatial region at the i^{th} channel is weighted by the j^{th} word in the description. Next, we further sum the word-weighted image features n at the C dimension to get $n' \in \mathbb{R}^L$, and normalize n' by the softmax function:

$$\delta_i = \frac{\exp(n'_i)}{\sum_{k=0}^{L-1} \exp(n'_k)}, \tag{5.2}$$

where δ_i reflects the correlation between the i^{th} word and the whole image. That is, it represents the chance that i^{th}-word-related visual attributes exist in the image. Thus, the word-level feedback used to train the generator can be calculated using Eq. 5.3:

$$\mathcal{L}_{\text{word}}(I, S) = \text{BCE}(\delta, l), \tag{5.3}$$

where BCE denotes binary cross-entropy, and l is the target label (see Section 5.1.2.2). By doing this, the discriminator provides the generator with explicit fine-grained

training feedback at word-level by examining whether word-related image attributes are generated in the synthetic image. This helps to disentangle different words in the sentence and visual attributes in the image, and enables an independent modification of each image regions according to the text descriptions. There are no trainable parameters in our word-level discriminator, and the implementation of it does not affect the context of the general description, as the objective functions have included the full sentence (S) as conditional adversarial losses to convey a rich context of text, ensuring that the generated images align with the semantic meaning of the sentence.

5.1.2.4 Generator of the Lightweight Architecture

Thanks to our word-level discriminator providing fine-grained training feedback related to each word, we are able to build a lightweight generator with a simple structure, as shown in Fig. 5.2. The generator consists of a text encoder, image encoders providing two different image features, and a series of upsampling and residual blocks, where the text encoder is a pre-trained bidirectional RNN (Xu et al., 2018), and the image encoders are pre-trained Inception-v3 (Szegedy et al., 2016) and VGG-16 (Simonyan and Zisserman, 2014) networks, respectively.

First, the generator encodes the input image I into image features using the Inception-v3 network, and then concatenates them with text features, encoded by the text encoder from the description S. Next, the image-text features are fed into a series of upsampling and residual blocks, followed by an image generation network to produce the modified result with the desired resolution.

We adopt conditioning augmentation (Zhang et al., 2017a) to smooth text representations, and text-image affine combination module (ACM) proposed in Section 4.2 to fuse text and image features. Note that for the last ACM, we use the VGG-16 network to extract image features, which contains more content details, helping rectify inappropriate attributes and complete missing contents. Besides, spatial attention (Xu et al., 2018) and channel-wise attention proposed in Section 4.1 are applied. Following ControlGAN (see Section 4.1), perceptual loss (Johnson et al., 2016) is adopted to reduce the randomness involved in the generation process and to help preserve text-irrelevant contents:

$$\mathcal{L}_{\text{per}}(I', I) = \frac{1}{C_i H_i W_i} \left\| \phi_i(I') - \phi_i(I) \right\|_2^2, \tag{5.4}$$

where I is the input image, I' is the modified result, $\phi_i(I)$ is the activation of the i^{th} layer of a pre-trained VGG-16 network, C_i is the number of channels, and H_i and W_i are the height and width of the feature map, respectively.

Why do we use two different image encoders? The image features extracted by the deeper Inception-v3 network are more semantic, as they primarily capture basic information of the input image, such as the layout, category, and rough shape of objects. These features are easier to interact with corresponding semantic words, enabling effective manipulation. Therefore, we utilize these coarse image features at the beginning and middle of the network to enhance the manipulation ability.

In contrast, the image features extracted by the shallower VGG-16 network contain numerous content details. These details may lead the generator to simply reconstruct the input image and limit the generation of new attributes specified in the text description. Consequently, we utilize these informative details towards the end of the model to correct modified text-irrelevant attributes and complete missing contents.

5.1.2.5 Objective Functions

Only a pair of generator and discriminator exists in our model, and we train the generator and the discriminator alternatively by minimizing both the generator loss $\mathcal{L}_G$ and the discriminator loss $\mathcal{L}_D$. Following ManiGAN, we use paired data $(I, S) \rightarrow I$ to train our model, where S is the text description matching the image I.

Generator Objective. The complete generator objective consists of unconditional and conditional adversarial losses, a perceptual loss $\mathcal{L}_{\text{per}}$ (Eq. 5.4), a word-level training loss $\mathcal{L}_{\text{word}}$ (Eq. 5.3), and a text-image matching loss $\mathcal{L}_{\text{DAMSM}}$.

$$\begin{aligned}\mathcal{L}_G = &\underbrace{-\frac{1}{2}E_{I'\sim P_G}\left[\log(D(I'))\right]}_{\text{unconditional adversarial loss}}\underbrace{-\frac{1}{2}E_{I'\sim P_G}\left[\log(D(I', S))\right]}_{\text{conditional adversarial loss}} \\ &+ \lambda_1\mathcal{L}_{\text{per}}(I', I) + \lambda_2\mathcal{L}_{\text{word}}(I', S) + \lambda_3\mathcal{L}_{\text{DAMSM}},\end{aligned} \tag{5.5}$$

where I is the real image sampled from the true image distribution P_{data}, I' is the generated image sampled from the model distribution P_G, λ_1, λ_2, and λ_3 are hyperparameters controlling different losses, and $\mathcal{L}_{\text{DAMSM}}$ (Xu et al., 2018) measures the text-image matching score based on the cosine similarity.

Discriminator Objective. The complete discriminator objective is defined as:

$$\begin{aligned}\mathcal{L}_D = &\underbrace{-\frac{1}{2}E_{I\sim P_{\text{data}}}\left[\log(D(I))\right] - \frac{1}{2}E_{I'\sim P_G}\left[\log(1 - D(I'))\right]}_{\text{unconditional adversarial loss}} \\ &\underbrace{-\frac{1}{2}E_{I\sim P_{\text{data}}}\left[\log(D(I, S))\right] - \frac{1}{2}E_{I'\sim P_G}\left[\log(1 - D(I', S))\right]}_{\text{conditional adversarial loss}} \\ &+ \lambda_4(\mathcal{L}_{\text{word}}(I, S) + \mathcal{L}_{\text{word}}(I', S)),\end{aligned} \tag{5.6}$$

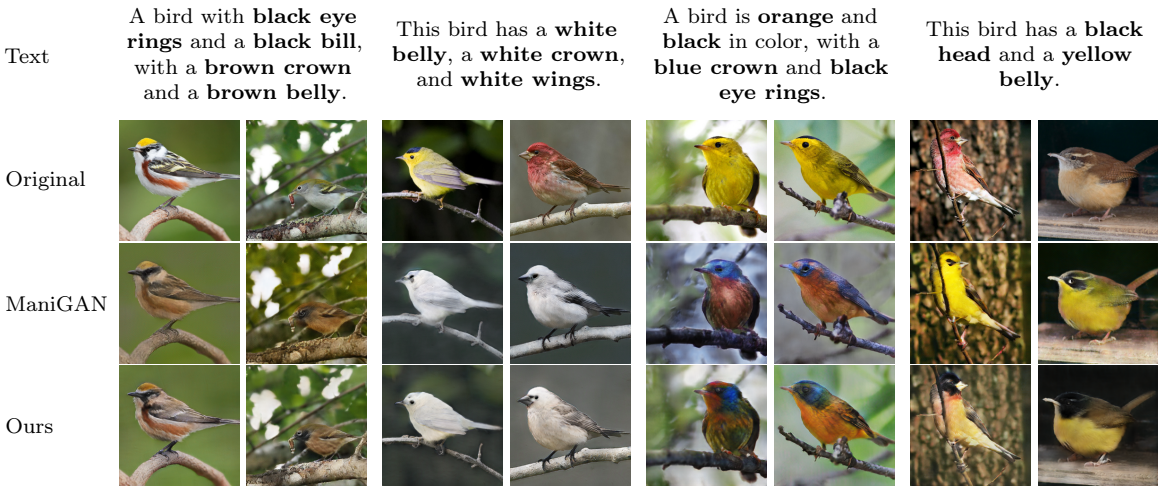


Figure 5.3: Qualitative comparison of two methods on the CUB.

where λ_4 is a hyperparameter. Note that the target label for this $\mathcal{L}_{\text{word}}(I', S)$ is set to 0 for all words.

5.1.3 Experiments

Table 5.1: Quantitative comparison: Fréchet inception distance (FID), accuracy, and realism of the state of the art and our method on the CUB and COCO. "w/o Dis." denotes without implementing proposed word-level discriminator. "w/ TAD" denotes using the text-adaptive discriminator (Nam et al., 2018) instead of our word-level discriminator. "w/ CD" denotes implementing the word-level discriminator introduced in ControlGAN. For FID, lower is better; for accuracy and realism, higher is better.

	CUB			COCO		
Method	FID	Accuracy	Realism	FID	Accuracy	Realism
ManiGAN	9.75	34.06	42.18	25.08	22.03	32.47
Ours	8.02	65.94	57.82	12.39	77.97	67.53
Ours w/o Dis.	9.20	-	-	15.60	-	-
Ours w/ TAD	8.84	-	-	14.95	-	-
Ours w/ CD	8.72	-	-	13.68	-	-

We evaluate our model on the CUB bird (Wah et al., 2011) and more complicated COCO (Lin et al., 2014) datasets, comparing with ManiGAN introduced in Section 4.2, which also focuses on text-guided image manipulation. Results for the method are reproduced using the code released by the authors.

Implementation. The scale of the output images is 256×256, but the size is adjustable to satisfy users' preferences. Similarly to ManiGAN, there is a trade-off

Table 5.2: Quantitative comparison: number of parameters in generator (NoP-G) and discriminator (NoP-D), runtime per epoch (RPE), and inference time for generating 100 new modified images (IT) of the state of the art and our method on the CUB and COCO. "Ours*" is the smallest number of parameters our model can have without significantly sacrificing image quality. For all evaluation matrices, lower is better. All methods are benchmarked on a single Quadro RTX 6000 GPU.

	CUB				COCO			
Method	NoP-G	NoP-D	RPE (h)	IT (s)	NoP-G	NoP-D	RPE (h)	IT (s)
ManiGAN	41.1M	169.4M	0.138	2.470	53.3M	377.6M	1.992	4.485
Ours	18.5M	71.8M	0.089	0.425	30.1M	160.7M	1.475	0.718
Ours*	5.4M	1.6M	0.058	0.356	7.4M	3.5M	0.760	0.445

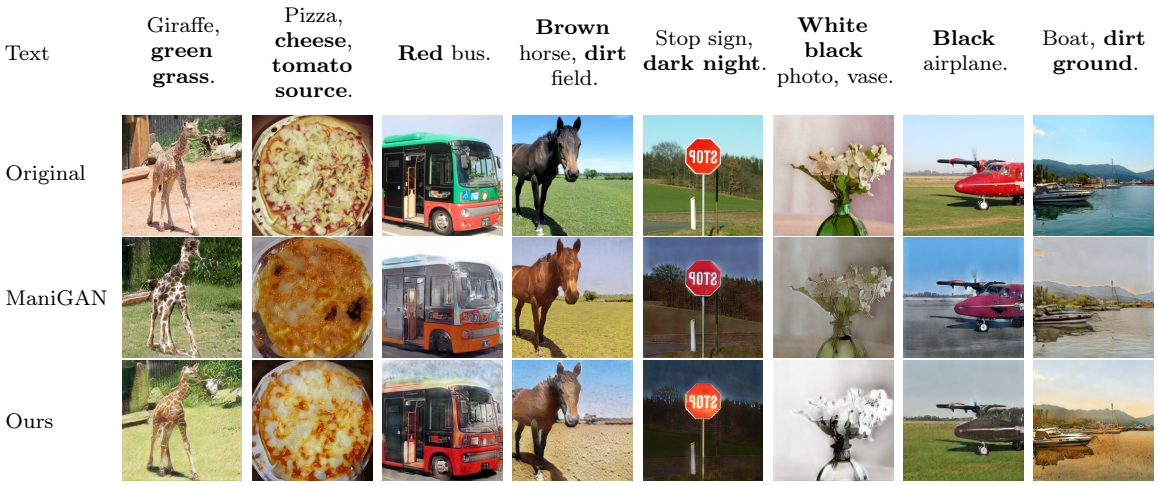


Figure 5.4: Qualitative comparison of two methods on the COCO.

between the generation of new attributes matching the text description and the preservation of text-irrelevant contents of the original image. Therefore, based on the manipulative precision (MP) proposed in Section 4.2, the whole model is trained 100 epochs on CUB and 10 epochs on COCO using the Adam optimizer (Kingma and Ba, 2014) with learning rate 0.0002. The hyperparameters λ_1, λ_2, λ_3, and λ_4 are all set to 1 for both datasets.

5.1.3.1 Comparison with State of the Art

Quantitative comparison. To evaluate the quality of synthetic images, the Fréchet inception distance (FID) (Heusel et al., 2017) is adopted as a quantitative evaluation measure. Also, to better verify the manipulation performance of our method, we conducted a user study. For each dataset, we randomly selected 30 images with

randomly chosen description (for COCO, each text-image pair is from the same category). Thus, there are in total 60 samples from two datasets for each method. Then, we asked workers to compare two results after looking at the input image, given text and outputs based on two criteria: (1) accuracy: whether the modified visual attributes of the synthetic image are aligned with the given description, and text-irrelevant contents are preserved, and (2) realism: whether the modified image looks realistic. Finally, we collected 1380 results from 23 workers, shown in Table 5.1. Besides, to evaluate the efficiency, we record the runtime for training a single optimization epoch (RPE) and the inference time (IT) for generating 100 new modified images. In the experiments, FID is evaluated on a large number of modified samples produced from mismatched pairs, i.e., randomly chosen input images edited by randomly selected text descriptions.

As shown in Table 5.1, compared with the state of the art, our method achieves better FID values on both CUB and COCO datasets, and both the accuracy and the realism show that our results are most preferred by workers. Besides, as shown in Table 5.2, the RPE and IT of our method are obviously smaller, and our method also has a much smaller number of parameters in both the generator and the discriminator when it has similar settings (e.g., size of hidden features) as the ManiGAN. Moreover, thanks to our powerful discriminator, we can even further reduce the number of parameters without sacrificing significant image quality. Table 5.2 "Ours*" records the smallest number of parameters that our model can have without significantly affecting the performance. Also, Fig. 5.5 plots the FID values corresponding to a different number of parameters in the generator and the discriminator.

This indicates that (1) our method can produce high-quality modified results with good diversity, (2) synthetic images are highly aligned with the given descriptions, while effectively preserving other text-irrelevant contents, and (3) our method is more friendly to memory-limited devices without sacrificing image quality.

Qualitative comparison. Figs. 5.3 and 5.4 show visual comparisons between ManiGAN and our method on CUB and COCO. As we can see, ManiGAN fails to produce satisfactory modified results on both datasets, and text-irrelevant contents are also changed. For example, in Fig. 5.3, the ManiGAN colors the branch using the color of bird, and in Fig. 5.4, it produces distorted objects (e.g., giraffe and pizza), and even fails to produce required attributes (e.g., color of airplane and style of the vase). We think the less satisfactory performance of ManiGAN is mainly because the discriminator fails to provide fine-grained training feedback related to each word.

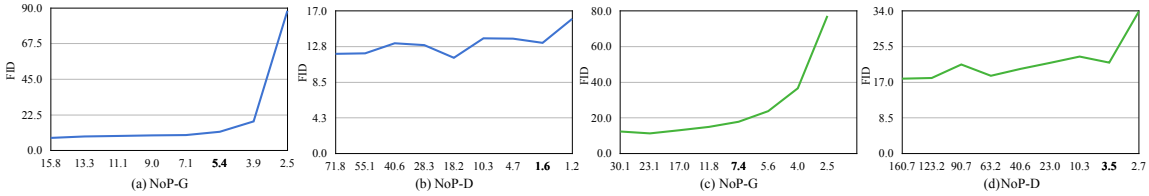


Figure 5.5: Fréchet inception distance (FID) at different numbers of parameters in the generator (or discriminator) when the number of parameters in the discriminator (or generator) is fixed. (a) and (b) are evaluated on the CUB dataset, and (c) and (d) are evaluated on the COCO dataset. In (a), the number of parameters in the discriminator is fixed at 71.8M; in (b), the number of parameters in the generator is fixed at 5.4M; in (c), the number of parameters in the discriminator is fixed at 160.7M; and in (d), the number of parameters in the generator is fixed at 7.4M.

5.1.3.2 Component Analysis

Effectiveness of the word-level discriminator. To verify the effectiveness of our proposed word-level discriminator, we conducted an ablation study shown in Figure 5.6 (c). As we can see, without the proposed word-level discriminator, the model fails to achieve effective manipulation on both datasets. For example, the model cannot produce a yellow head and a yellow belly of the bird, and branches are colored red and black as well. This indicates that due to the lack of word-level training feedback, the generator fails to disentangle different visual attributes and cannot build an appropriate word-region connection to achieve effective modification. Furthermore, FID values shown in Table 5.1 further verify the effectiveness of our proposed discriminator, as the values deteriorate significantly when the model does not implement the proposed word-level discriminator.

Comparison between different word-level discriminators. To verify the superiority of our word-level discriminator, we conducted a comparison study between the text-adaptive discriminator (Nam et al., 2018), the word-level discriminator introduced in ControlGAN (see Section 4.1), and our word-level discriminator, shown in Fig. 5.6 (d) and (e). The model with the text-adaptive discriminator (d) changes text-irrelevant contents (e.g., the background), which is mainly due to the loss of spatial information in the discriminator. Also, the model with the word-level discriminator introduced in ControlGAN (e) fails to produce the required attributes (e.g., the red bird without yellow parts and the zebra without the blue sky), and some contents that are not described in the text are modified as well (e.g., the white and black branch), which is mainly caused by the coarse training feedback provided by the discriminator.

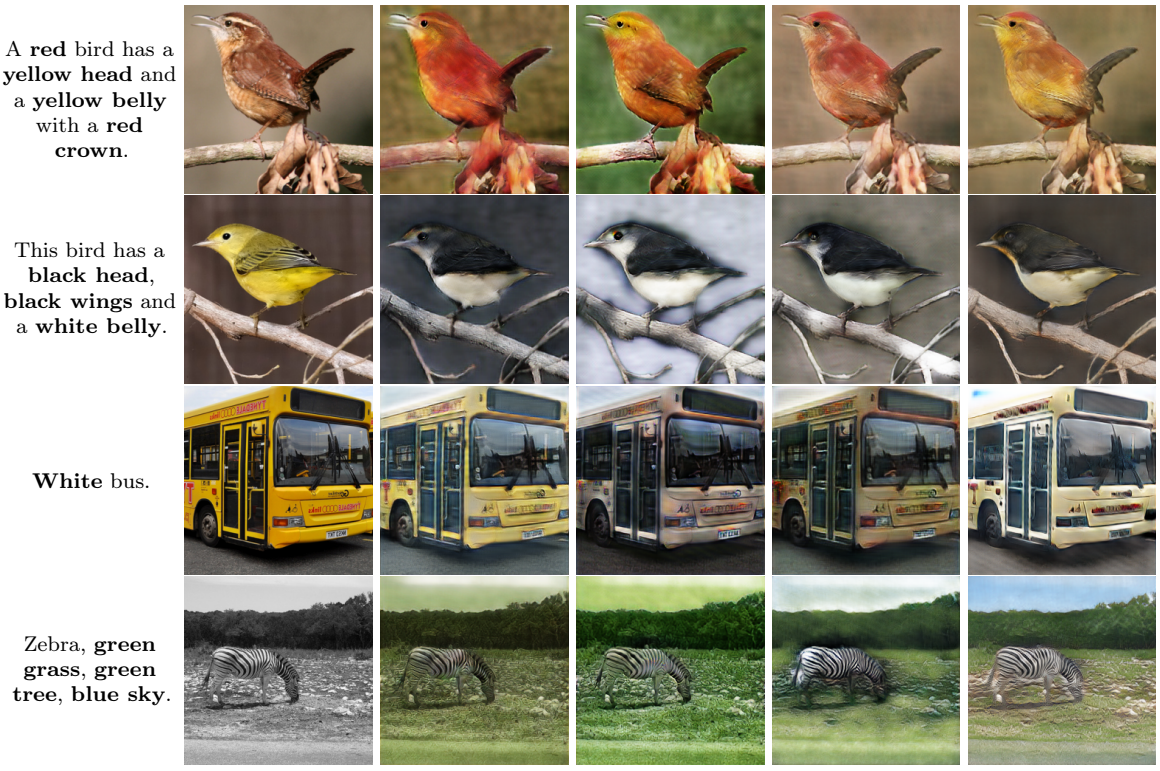


Figure 5.6: Ablation and comparison studies. a: text description with desired visual attributes; b: input image; c: removing the proposed word-level discriminator; d: replacing our discriminator with the text-adaptive discriminator (Nam et al., 2018); e: replacing our discriminator with the word-level discriminator used in ControlGAN proposed in Section 4.1 f: our full model.

Visualization of attention maps. To verify the effectiveness of our word-level discriminator, we visualize attention maps produced by models with/without different word-level discriminators, as shown in Fig. 5.7. We can observe that the model with our proposed word-level discriminator can generate a better spatial attention map with more accurate locations, finer shapes aligned with objects, and better semantic consistency between visual attributes and corresponding words. On the contrary, models with the other two word-level discriminators fail to generate appropriate attention maps or only produce coarse attention maps in the wrong location.

5.1.4 Conclusion

We have proposed a novel and powerful word-level discriminator along with explicit word-level supervisory labels, which can provide the generator with fine-grained training feedback related to each word, and thus enable the construction of a lightweight

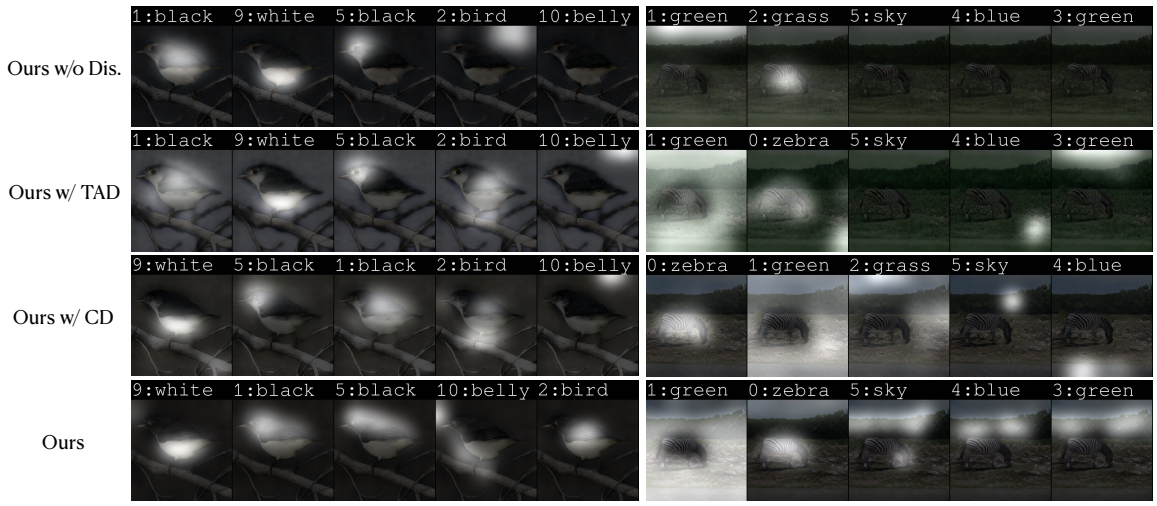


Figure 5.7: Visualization of attention maps that are produced by models with/without different word-level discriminators.

architecture for image manipulation using natural language descriptions. More specifically, compared with the state of the art, our method has a much smaller number of parameters, but still achieves a competitive manipulation performance, which is friendly enough to memory-limited devices. Extensive experimental results demonstrate the efficiency and superiority of our method on two benchmark datasets.

5.2 Single-Direction Discriminator

In this section, we rethink the purpose of the discriminator in conditional generative adversarial networks. Differently from the previous section, which aimed to improve the capabilities of the discriminator to enable a reduction in the number of trainable parameters in a generative network, we introduce a *single-direction* discriminator that combines two training goals (i.e., having both good image quality and text-image alignment) into one direction (i.e., improving the quality of fusion features). To evaluate the performance of our proposed discriminator, we adopt a story visualization task. Additionally, we introduce a new sentence representation with word information and an extension of word-level spatial attention to work as complements to our proposed single-direction discriminator.

5.2.1 Discriminator with Fusion Features

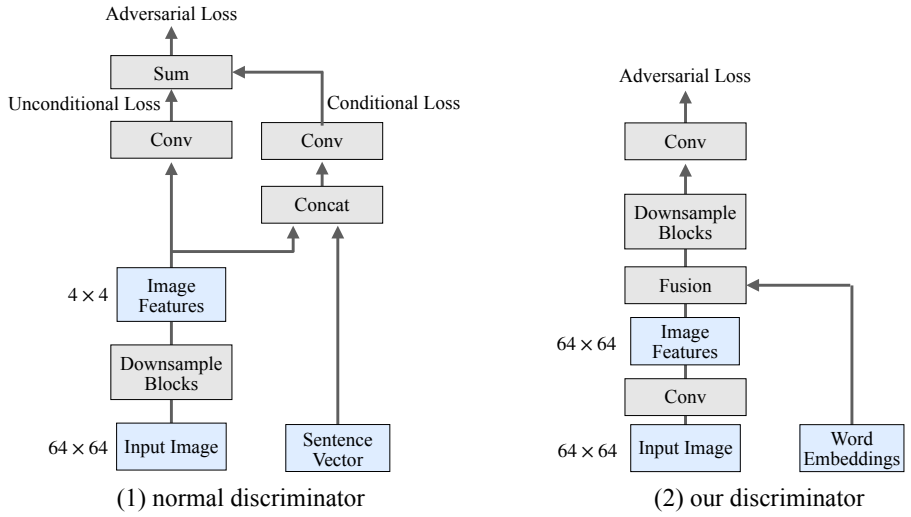


Figure 5.8: Comparison between the normal two-pipeline discriminator (1) and our single-pipeline discriminator (2), where the normal discriminator needs to check image quality and text-image alignment, while our discriminator only needs to check the quality of fusion features.

In this work, we propose a new single-pipeline discriminator, which works on fusion features that contain both image and text information. Although DFGAN (Tao et al., 2020) also introduced an single-pipeline discriminator, differently, (1) our discriminator uses fine-grained word-level information and image features at the original size without information loss, while DFGAN first downsamples image features to a small size at 4×4,

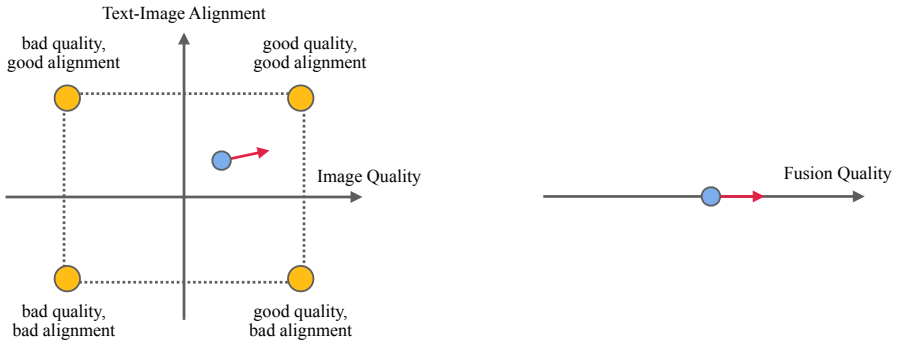


Figure 5.9: A diagram for two training goals of the normal discriminator (1) and the goal of our proposed single-pipeline discriminator (2).

which may suffer a potential information loss problem, (2) our discriminator builds fusion features via applying matrix multiplication between detailed image features and word-level text representations, while DFGAN concatenates a coarse sentence vector with small-scale image features at 4×4, and may fail to comprehensively explore the correlation between image and text representations, and (3) the loss objectives for our discriminator are only to check the difference of fusion features created from word representations and synthetic images and the fusion features created from word representations and real images. In contrast, the fusion features in DFGAN serve to promote an additional gradient penalty, along with the discriminator objectives.

Fusion Features. As shown in Fig. 5.8, (1), in current methods (Xu et al., 2018; Qiao et al., 2019b), the discriminator first extracts image features with a small size at 4×4 through a series of downsampling blocks. Then, this small-size features are used in two ways: one way is to determine whether the image is real or fake, and the other way is to concatenate this image features with the sentence vector to evaluate text-image semantic consistency. So, there are two kinds of loss computed, the unconditional adversarial loss and the conditional adversarial loss.

However, concatenating such small-size image features with the coarse sentence vector may not fully evaluate the text-image semantic alignment, because the image features at 4×4 may lose much text-matched image information, and the sentence vector is a global representation of a given text, and cannot comprehensively reflect the alignment between image and text. Based on this, to better evaluate the text-image semantic consistency, we suggest to utilize word embeddings, and combine both text and image information at a shallow layer of the discriminator, instead of at size 4×4.

Here, we introduce fusion features, which contain fine-grained text information at word-level and image information at a larger resolution, and then feed this detailed fusion features into the discriminator to evaluate the text-image semantic consistency (see Fig. 5.8, (2)). Basically, checking the difference between fusion features created from word representations and synthetic images and fusion features created from word representations and real images can also reflect the realism of synthetic images. Good fusion features should be both realistic and have good text-image semantic alignment, just like real images matching their corresponding text descriptions.

To build this fusion features, we convert the output real/fake image features $v \in \mathbb{R}^{C\times H\times W}$ into the same semantic space as word embeddings w, to get $v' \in \mathbb{R}^{D\times(H*W)}$. Then, the fusion features $F \in \mathbb{R}^{L\times H\times W}$ are obtained by applying a matrix multiplication between v' and w, denoted as $F = \text{Reshape}((w)^T v')$, where each value in F denotes the correlation between each pixel and each word. Finally, we feed fusion features into the discriminator.

Single-Pipeline Discriminator. The discriminator in conditional generative adversarial networks computes two types of losses: unconditional adversarial loss for image quality and conditional adversarial loss for text-image semantic alignment, as shown in Fig. 5.8 (1). These losses promote the generator to achieve two goals: producing synthetic images that are indistinguishable from real images and capturing semantic alignment between synthetic images and text as effectively as real images and text. However, as these goals can be treated independently, the network may fail to achieve both simultaneously. The implementation of fusion features in our discriminator helps combine these goals into one, promoting the generator to produce fusion features that are indistinguishable from those built from real images and text, as shown in Fig. 5.9 (4). Building good fusion features should result in high-quality output images and effective correlation between image regions and corresponding word information. Therefore, our loss functions aim to train the model to produce such fusion features.

$$
\begin{aligned}
\mathcal{L}_{G,I} &= -E_{F'_I\sim P_g}\left[\log(D(F'_I))\right], \\
\mathcal{L}_{D,I} &= -E\left[\log(D(F_I))\right] - E_{F'_I\sim P_g}\left[\log(1-D(F'_I))\right], \\
\mathcal{L}_{G,V} &= -E_{F'_V\sim P_g}\left[\log(D(F'_V))\right], \\
\mathcal{L}_{D,V} &= -E\left[\log(D(F_V))\right] - E_{F'_V\sim P_g}\left[\log(1-D(F'_V))\right],
\end{aligned}
\tag{5.7}
$$

where I denotes an image, V denotes a story with a series of images, F are real fusion features created by word representations and real image (or story) features that are sampled from the real distribution, F' are synthetic fusion features created by word

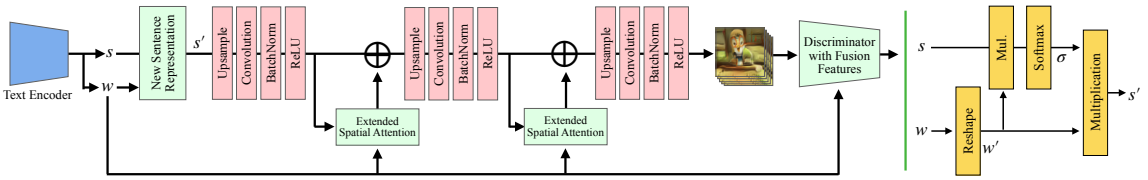


Figure 5.10: Left: architecture of the proposed approach. Right: Design of the proposed new sentence representation.

representations and synthetic image (or story) features that are sampled from the model distribution.

5.2.2 Another Components

The complete architecture is shown in Fig. 5.10, left. In addition to our proposed single-pipeline discriminator, we introduce two additional components in the network. These components are designed to complement the functionality of the discriminator and leverage the fine-grained training feedback it provides. The integration of these components enables the construction of a lightweight architecture.

5.2.2.1 New Sentence Representation

One problem arising in previous works (Li et al., 2019c; Song et al., 2020) is that they feed given story sentences sequentially into the generation pipeline to produce corresponding story images. Thus, when some keywords do not appear in all sentences (e.g., as shown in Fig. 5.12, the word "tent" only appear in the fifth sentence), the corresponding keyword-related attributes may not be generated in all story images due to the sequential generation, and then the synthetic story may fail to keep consistency across all images.

To address the issue, we propose a new sentence representation, shown in Fig. 5.10, right, which can selectively incorporate different word information from the entire story to mitigate the inconsistency problem. To build this new sentence representation, we first reshape the word embeddings $w \in \mathbb{R}^{N \times D \times L}$ to get $w' \in \mathbb{R}^{D \times (L*N)}$. Then, we calculate the correlation weights $\sigma \in \mathbb{R}^{N \times (L*N)}$ between each sentence and all words within the whole story, according to applying a matrix multiplication followed by a Softmax operation between w' and a sentence vector $s \in \mathbb{R}^{N \times D}$, denoted as $\sigma = \text{Softmax}(sw')$. Next, a new sentence representation weighted by word-sentence relation can be obtained by doing $s' = \sigma(w')^T$, and is fed into the generation pipeline at the beginning. By doing this, each sentence vector of s' captures the relationship between the sentence and the individual words. This enables the network to learn how

to incorporate different word information from the story, even if certain words only appear in specific sentences. The goal is to maintain information consistency across all story images, ensuring that the generated images align with the overall narrative conveyed by the text.

5.2.2.2 Extended Word-Level Spatial Attention

To generate high-quality story images with finer details and a better consistency, we adopt word-level spatial attention (Xu et al., 2018), and further extend it to focus on all words and visual spatial locations in the entire story. By doing this, our extended attention not only highlights local word-related image regions in the generation process, but also captures the long-range correlation between words in the current sentence and visual spatial locations from other images within the same story, which ensures both the local image region quality and the global consistency of the whole story, tailored for the story visualization task.

First, we convert intermediate visual features $v \in \mathbb{R}^{N\times C\times H\times W}$ into a joint semantic space $\mathbb{R}^D$ via a convolution layer, and then reshape the new features to get $v' \in \mathbb{R}^{D\times(H*W*N)}$, where N is the size of a story, C is channel number, H is height of the features, and W is width. We also reshape word embeddings $w \in \mathbb{R}^{N\times D\times L}$ into $w' \in \mathbb{R}^{D\times(L*N)}$, where D is the feature dimension and L is the number of words in a sentence. Then, the extended spatial attention $\beta \in \mathbb{R}^{(H*W*N)\times(L*N)}$ focusing on capturing correlation between words and visual spatial locations across the entire story can be obtained by:

$$\beta_{i,j} = \frac{\exp(a_{i,j})}{\sum_{l=0}^{L*N-1}\exp(a_{i,l})}, \quad \text{where } a = v'^T w', \tag{5.8}$$

where $\beta_{i,j}$ denotes the correlation between the ith spatial location and the jth word in the story. Therefore, weighted visual features containing word information v_w can be obtained by $v_w = w'\beta^T$. By doing this, the model encourages all images within a story not only to have fine-grained regional details matching corresponding semantic words, but also share similar appearances to make the whole story more consistent.

5.2.3 Experiments

There are limited number of methods working on the same story visualization task as ours: StoryGAN (Li et al., 2019c), CP-CSV (Song et al., 2020), DUCO (Maharana et al., 2021), and VLC (Maharana and Bansal, 2021). StoryGAN is based on generative adversarial networks, and CP-CSV, DUCO, and VLC are built on top of StoryGAN,

Table 5.3: Quantitative evaluation between different methods on the Pororo-SV and Abstract Scenes. For FID and FSD, lower is better; for text-image cosine similarity (Cosine), higher is better.

	Pororo-SV dataset			Abstract dataset		
Method	FID↓	FSD↓	Cosine↑	FID↓	FSD↓	Cosine↑
StoryGAN	78.64	94.53	0.22	135.16	55.80	3.59
CP-CSV	67.76	71.51	0.32	-	-	-
DUCO	95.17	171.70	0.08	142.34	49.16	3.95
VLC	94.30	122.07	0.21	-	-	-
Ours	**56.08**	**52.50**	**2.98**	**72.34**	**14.86**	**4.05**

where CP-CSV relies on character segmentation masks to provide an additional supervision, and DUCO and VLC utilized auxiliary captioning networks to further keep text-image semantic alignment.

5.2.3.1 Implementation

Our approach is developed using PyTorch, building on top of the original StoryGAN codebase. The resolution of output images on Pororo-SV is 64×64, and on Abstract Scenes is 256×256. The text encoder is a bi-directional LSTM, pre-trained to maximize the cosine similarity between matched image and text features (Xu et al., 2018). We select the best checkpoints and tune hyperparameters by using the FID and FSD scores mentioned below. The network is trained 240 epochs on both Pororo-SV and Abstract Scenes, using the Adam optimizer (Kingma and Ba, 2014) with learning rate 0.0002. All models were trained on a single Quadro RTX 6000 GPU.

5.2.3.2 Quantitative Evaluation

Table 5.3 shows quantitative comparisons between our approach and the baselines on Pororo-SV and Abstract Scenes. From the table, we can observe that our approach achieves better evaluation results against other methods on both datasets, and builds a new state of the art. Note that differently from CP-CSV, DUCO, and VLC, our approach uses neither segmentation masks nor auxiliary captioning networks. Compared to StoryGAN, our approach achieves 28.7% improvement in FID, and 44.5% improvement in FSD on Pororo-SV; and as for Abstract Scenes, our approach achieves 46.5% improvement in FID, and 73.4% improvement in FSD. This illustrates



Figure 5.11: Comparison between different methods on the Pororo-SV.

that our approach can generate images with finer quality, achieve a better image-text semantic alignment, and keep a higher consistency across story images. Note that we do not evaluate CP-CSV and VLC on Abstract Scenes, as CP-CSV requires segmentation masks, and VLC has not released the complete code.

5.2.3.3 Qualitative Evaluation

Figs. 5.11 and 5.12 present visual comparisons between our approach and the baselines on the Pororo-SV and Abstract Scenes datasets, respectively. The baselines on Pororo-SV exhibit low-quality images with distorted characters and blurred backgrounds. For instance, the penguin Pororo generated by StoryGAN lacks recognizable details. In contrast, our approach produces images with finer regional details, including accurate character appearances (e.g., penguin Pororo and frog Crong) and sharper object shapes (e.g., glasses and hat on penguin Pororo).

For Abstract Scenes, the baselines may fail to keep the characters consistent or maintain text-image alignment, e.g., DUCO generates two boys Mike or two girls Jenny in a story image. On the other hand, our approach produces high-quality characters (e.g., boy Mike and girl Jenny) with complete and detailed appearances, and also ensures better consistency. For example, the word "tent" only appears in the first and last sentences of the second example, but all synthetic images in the story

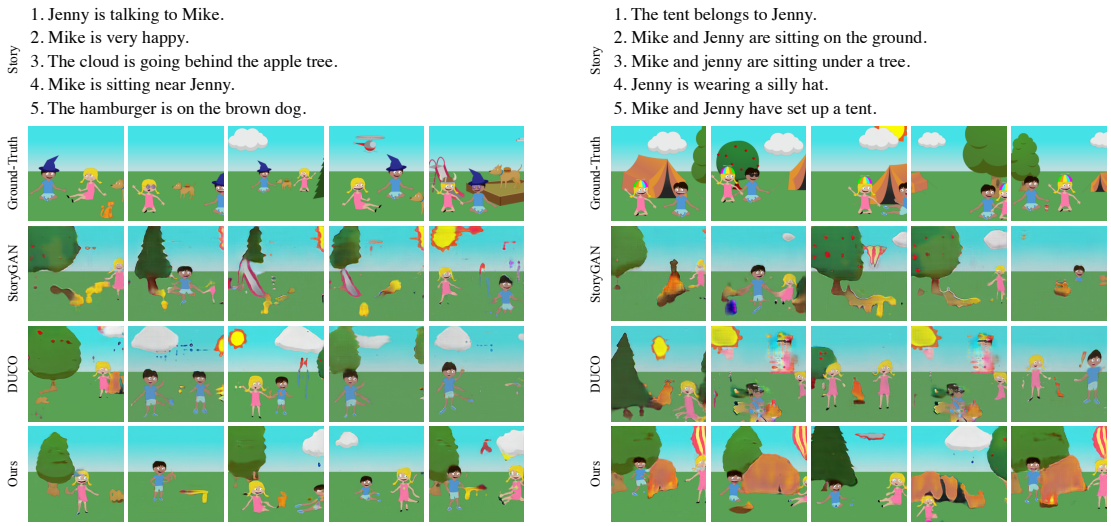


Figure 5.12: Comparison between different methods on the Abstract Scenes.

have the object tent. Other methods fail to keep the object tent in the background of all images within the same story.

5.2.3.4 Component Analysis

An ablation study to evaluate the effectiveness of different components proposed in the paper, and the results are shown in Table 5.4.

Sentence Representation with Word Information. First, we find that the new sentence representation improves scores on all evaluation metrics. We attribute this improvement to the better representation of text that contains fine-grained word information from the entire story, which enables high-quality initial image features with finer regional details and a better consistency (e.g., better FSD score), and thus further improves final synthetic results (e.g., better FID and Cosine scores). This is also supported by the observation found in text-to-image generation (Zhu et al., 2019), where the quality of initial image features can considerably affect the quality of output images in a sequential upsampling generation pipeline (see Fig. 5.10).

Discriminator with Fusion Features and Extended Word-Level Spatial Attention. A degradation can be observed when our approach does not implement the proposed discriminator. Similarly, without adopting proposed extended spatial attention, we can observe a decrease in both FID and FSD scores. We think that this is because both components are complement to each other: (1) without the proposed discriminator, although the generator tries to make use of word-level information via

Table 5.4: Component Analysis on the Pororo-SV. "Ours w/o Sentence" stands for without using the proposed new sentence representation; "Ours w/ Discrimiantor" stands for using the discriminator in current story visualization methods; "Ours w/o Extended Spatial Attention" stands for without adopting the proposed attention; "Ours w/ Word-Level Spatial Attention" is with the implementation of word-level spatial attention (Xu et al., 2018), instead of our proposed extended spatial attention.

Method	FID	FSD	Cosine
Ours w/o New Sentence Representation	68.48	62.85	2.24
Ours w/ Discriminator	62.23	59.33	2.61
Ours w/o Extended Spatial Attention	83.66	78.80	2.26
Ours w/ Word-Level Spatial Attention	58.26	63.39	2.54
Ours w/ Pre-trained BERT	52.38	49.69	3.71
Ours w/ FT BERT	50.96	48.81	3.95
Ours w/ BERT Scratch	55.78	51.71	2.80
Ours	56.08	52.50	2.98

the proposed attention in the generation process, there is no corresponding word-level feedback from the discriminator, regarding whether word-related attributes are produced in synthetic story images, and (2) without the extended spatial attention, the fine-grained feedback from the proposed discriminator at both word- and image region-level cannot be fully utilized by the generator. Thus, both components work together to make a full use of word information to enable a higher-quality image generation with a better story consistency. Besides, we find that the scores for "Ours w/o New Sentence Representation" is better than "Ours w/o Extended Spatial Attention". This shows that the fine-grained training feedback provided by our discriminator can be more utilized by the proposed extended word-level spatial attention.

Besides, we further record the number of trainable parameters in both kinds of discriminators, where our proposed single-direction discriminator only contains 10.4M trainable parameters, while the number of parameters in StoryGAN, DUCO, and VLC is 47.2M, and 70.9M for CP-CSV. So, compared to StoryGAN, our discriminator reduces about 77.97% number of parameters, but achieves 28.7% improvement in FID, and 44.5% improvement in FSD on Pororo-SV, which verifies its superiority.

Moreover, we replace the extended spatial attention with word-level spatial attention (Xu et al., 2018) to verify its effectiveness. The main improvement using the extended spatial attention is shown on the consistency of output results, especially on FSD score. This is because our proposed component focuses mainly on the whole story with multiple images and sentences, instead of only a pair of image and sentence, which coincides with the purpose of keeping the consistency across story images.

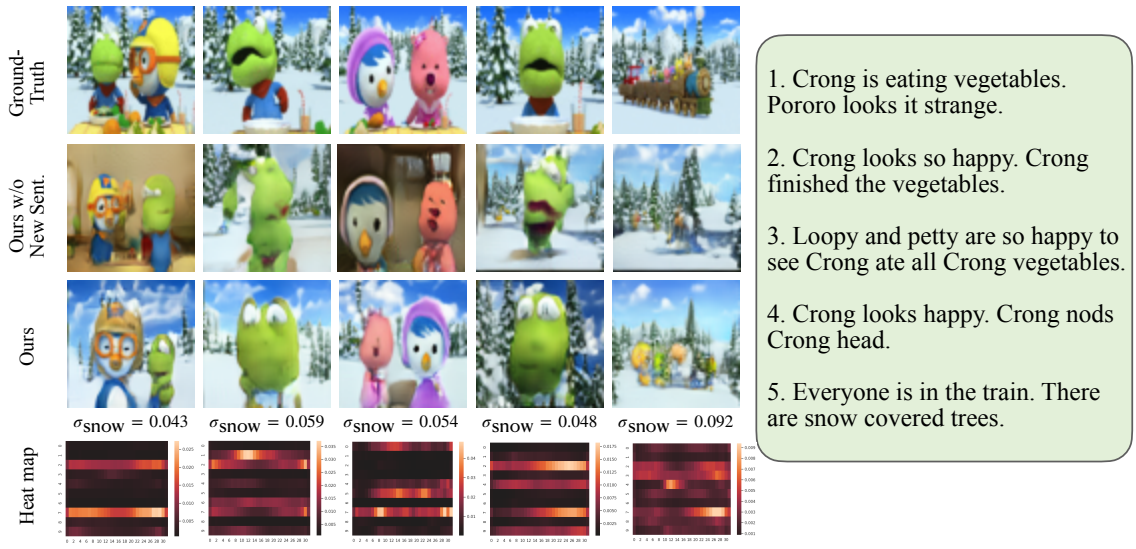


Figure 5.13: Visualization of the proposed extended spatial attention and new sentence representation.

BERT. In Table 5.4, we use BERT to replace LSTM as the text encoder. If the BERT is trained from scratch, denoted as "w/ BERT Scratch", there is no significant changes on all metrics. This is because Pororo-SV is small and each caption contains limited number of words, and thus a RNN-based text encoder (e.g., LSTM) is strong enough to capture long-range dependencies between words. If we use a pre-trained BERT, denoted as "w/ Pre-trained BERT" or further fine tune BERT on Pororo-SV, denoted as "w/ FT BERT", performance improves. This is because BERT pre-trained on a larger text corpus (e.g., Wikipedia) can generate better text representations.

5.2.3.5 Visualization

Capturing relations between words and image regions has been adopted in text-to-image generation (Xu et al., 2018; Qiao et al., 2019b) to improve image quality and text-image semantic alignment. Our proposed extended word-level spatial attention captures relations between image regions and words from the entire story to ensure local image quality and also global consistency across story images, tailored for story visualization. In Fig. 5.13, we visualize the extended spatial attention using heatmap. The y axis denotes the last 10 words in the story (i.e., sentence 5: Everyone...), and x denotes image regions, where we evenly split an image into 32 regions and compute the average for each region. So, the heatmap shows the correlation values captured by our attention between the last 10 words and image regions in each image. The word snow ($y = 7$) is highlighted in each heatmap, and snow is also generated in all

Table 5.5: Effects of our new sentence representation adopted in story visualization on the Pororo-SV and text-to-image generation on the CUB. For FID and FSD, lower is better, for IS, higher is better. "+ New Sent." means using our new representation.

Method	FID	FSD	IS
StoryGAN + New Sent.	72.81	84.06	-
CP-CSV + New Sent.	63.12	64.29	-
DUCO + New Sent.	87.82	131.83	-
VLC + New Sent.	82.19	100.94	-
AttnGAN	23.98	-	4.36
AttnGAN + New Sent.	19.20	-	4.71
DFGAN	14.81	-	5.10
DFGAN + New Sent.	11.98	-	5.16
Ours	56.08	52.50	-

output images, which verifies that our method captures distant word-image relations and ensures consistency even when some keywords do not appear in all sentences.

Furthermore, we also visualize the proposed new sentence representation with word information by showing the σ values between "snow" and each sentence in Fig. 5.13. As we can see, our sentence representation learns to give high σ to keywords, without it, distant images from keywords may not keep consistency (e.g., there is no snow in the 1st image of "Ours w/o New Sent.").

5.2.3.6 General Applicability of New Sentence Representation

In Table 5.5, we apply our new sentence representation to other story visualization methods as well as text-to-image generation approaches (AttnGAN (Xu et al., 2018) and DFGAN (Tao et al., 2020)) by incorporating word information into the sentence vector without modifying the networks themselves. We can observe that our sentence representation further enhances these methods, resulting in improved evaluation scores. In the case of story visualization, the improvement is mainly attributed to our proposed sentence representation, which enables high-quality initial image features with finer regional details and better consistency, leading to improved synthetic results. Regarding text-to-image generation, the improvement validates the observation presented in the work of Zhu et al. (2019) that the quality of initial image features impacts the quality of output images in a sequential upsampling generation pipeline.

5.2.3.7 Human Evaluation

We also conduct a human evaluation to comprehensively evaluate the performance of our approach on Pororo-SV. To do this, following Li et al. (2019c), a side-by-side

Table 5.6: Results of a human evaluation between Ours and DUCO on the Pororo-SV.

Choice (%)	Ours	Tie	DUCO
Visual Quality	75.3	8.3	16.4
Consistency	81.8	4.4	13.8
Relevance	72.3	5.7	22.0

human evaluation study is conducted based on three evaluation criteria: (1) the visual image quality, (2) the consistency among story images, and (3) the semantic alignment between paired image and sentence. The study compares the synthetic image sequences from our approach and DUCO. We showed the generated story images along with the corresponding story sentences from two methods, and three options are provided: (1) the first is better, (2) two are similar, and (3) the second is better. In this evaluation, we randomly choose 200 synthetic story images, generated from corresponding story descriptions sampled from the testing dataset, and each was assigned to 5 workers to reduce human variance.

From Table 5.6, story images produced by our approach are more preferred by workers on three evaluate criteria, which demonstrates the effectiveness of our approach on high-quality image generation with a better text-image semantic alignment, and a better story consistency.

5.2.4 Conclusion

In this paper, we investigated the task of story visualization. We proposed a new approach by exploring fine-grained information at both word and story level, and demonstrate the effectiveness of proposed components by conducting extensive experiments. Our approach has three novel components: a new sentence representation selectively incorporates different word information from the entire story to enable a better consistency, the extended spatial attention captures relations between image regions and words across the entire story to enable the generation of higher-quality story images with better consistency, and the novel discriminator contains a smaller number of parameters, but effectively provides the generator with fine-grained training signals, evaluating the quality of fusion features to examine whether word-related image attributes are generated in story images and also their quality, and thus encouraging the generator with the proposed attention to highlight image relations corresponding to semantic words from the story. Finally, a human evaluation has been conducted to further verify the effective performance of our proposed approach.

However, there is still much space for quality improvement on different datasets, such as the images details on characters. Also, a more concise architecture design is desired, as current methods (including our approach) require a large amount of memory storage to hold and train. Also, the resolution of output images is small at 64×64, which is less practical for real-word applications.

Chapter 6

General Conclusion and Perspective

In this thesis, I investigated three directions to effectively incorporate multi-modal information: (1) using a consistent representation for information with similar semantic meaning, (2) exploring correlations between image visual attributes and corresponding semantic words, and (3) developing lightweight architectures for models with inputs from multiple domains. For exploring correlations, I further investigated three approaches: word-level attention-based connections aided by a complementary word-level discriminator, text-image affine combinations, and a semi-parametric memory-driven approach. For lightweight architecture construction, two further directions were explored: improving the capabilities of generator and discriminator in conditional GANs to avoid blindly increasing network size, and constructing a single-directional discriminator by combining two training goals (i.e., better image quality and text-image semantic alignment) into a single direction (improving the quality of fusion features) to reduce redundancies in conditional GANs. The results are presented as follows:

First, having a consistent representation for information from different modalities with the similar semantic meaning can ensure that similar instances can be closely clustered within the same cluster, which allows the network to easily correlate this information, and then to achieve an effective translation or fusion between them without introducing ambiguities in the generative module. Combining both contrastive learning and clustering learning is an effective approach to achieve this, where both are complementary to each other, as different contrastive losses maximize the mutual information at different representation levels, and clustering losses help to capture semantic similarity and to prevent contrastive learning pushing instances with the similar semantic meaning far away from each other.

Future work on achieving a consistent representation for information with similar semantic meaning will be necessary to evaluate the approach in various tasks. This is because additional experiments on the general applicability of the proposed approach

would be valuable in providing a complete understanding of the need for a consistent representation in tasks that require multi-modal information as input. Furthermore, further investigation into the principles of both contrastive and clustering learning could facilitate a comprehensive implementation of both in multi-modal interaction. Additionally, since both contrastive and clustering learning can be computationally expensive, further exploration of alternative lightweight and effective approaches would be valuable.

Second, attention-based correlation enables the network to learn to correlate image visual attributes with corresponding semantic words from a given sentence, allowing the network to identify the semantic meaning of the text and specific visual attributes of the image regions. Then, affine combination is a simple but effective way to fuse multi-modal information, enabling the network to selectively choose required semantic word information and image visual attributes to achieve an effective interaction. Additionally, the proposed semi-parametric memory-driven approach combines the advantages of both parametric and non-parametric techniques. The non-parametric component is a memory bank of disentangled image features constructed from the training set of real images, and the parametric component is a generative adversarial network. At inference time, given a text description, 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 produce realistic synthetic results.

As future work, the three proposed approaches for correlating image visual attributes with corresponding semantic words can be further improved by addressing the semantic misalignment problem. This is because current methods typically use a fixed text encoder and image encoder to extract corresponding features and then use these features directly in a fixed GAN-based network. However, text representations can be coarse sentence vectors, fine-grained word embeddings, or structured knowledge, while image features extracted from different layers of a convolutional neural network (e.g., VGG) may contain different-level semantic information, as demonstrated by Gatys et al. (2016) and Johnson et al. (2016). Simply fusing cross-domain representations together using a fixed generative network may have a significant adverse impact on the quality of output images. For example, one of the primary functions of the discriminator in conditional GANs is to evaluate the semantic alignment between input text and output image and provide corresponding feedback to the generator, which encourages the generator to generate text-semantic-matched images. However, current methods only concatenate a coarse sentence vector and image features at a

small scale (i.e., 4x4), extracted from a given image using a series of convolutional layers, to evaluate the semantic alignment. This approach may fail to fully match the semantics between text and image features and provide less precise training feedback to the generator.

Furthermore, exploring more effective ways to build a memory bank, such as implementing more efficient data structures or faster search algorithms to accelerate the retrieval process of image features, would be another valuable direction for future work. This is because the proposed method currently needs to search through all stored information to find the most compatible features. Additionally, improving the similarity matrix used to evaluate the correlation between multi-modal information would be desirable for achieving a more comprehensive evaluation.

Third, according to improving the capabilities of both generator and discriminator, we are allowed to build a lightweight network having inputs from different modalities, which has a smaller number of parameters but with a competitive performance, compared to the method according to blindly increasing the size of a network. Also, current GAN-based networks may still have some redundancy in their architecture, and we find that having a single-directional discriminator is powerful enough for conditional GANs to produce high-quality synthetic images with a good text-image semantic alignment, instead of evaluating image quality and text-image semantic alignment separately by having two networks in the discriminator architecture.

Future work could further explore a lightweight architecture to advance the implementation of a network having inputs from multiple modalities in real-world applications. The two approaches proposed in this thesis need to be further studied by improving the capabilities of existing networks and rethinking the principle of each component to reduce unnecessary redundancies.

In summary, having a framework to achieve an effective interaction between multi-modal information can greatly improve our world, from allowing non-artists to generate visual appealing artworks, to achieving an effective information translation and propagation among us. My hope is that, in the future, we will have robust and accurate neural models that break various information gaps and freely allow all of us to understand different information and express ourselves.

Bibliography

Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *Advances in Neural Information Processing Systems*, 33:9758–9770, 2020.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

Yuki Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. *Advances in Neural Information Processing Systems*, 33:4660–4671, 2020.

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019.

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.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Richard Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3286–3295, 2019.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. "O'Reilly Media, Inc.", 2009.

Andrew Brock, Theodore Lim, James M. Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision*, pages 132–149. Springer, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

Brian Chen, Andrew Rouditchenko, Kevin Duarte, Hilde Kuehne, Samuel Thomas, Angie Boggust, Rameswar Panda, Brian Kingsbury, Rogerio Feris, David Harwath, et al. Multimodal clustering networks for self-supervised learning from unlabeled videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8012–8021, 2021.

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

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022.

Marleen De Bruijne. Machine learning approaches in medical image analysis: From detection to diagnosis, 2016.

Harm De Vries, Florian Strub, Jérémie Mary, Hugo Larochelle, Olivier Pietquin, and Aaron C. Courville. Modulating early visual processing by language. In *Advances in Neural Information Processing Systems*, pages 6594–6604, 2017.

Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3D imitative-contrastive learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5154–5163, 2020.

Emily L. Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a Laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1486–1494, 2015.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.

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.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. Text-based editing of talking-head video. *ACM Transactions on Graphics*, 38(4): 1–14, 2019.

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.

Jon Gauthier. Conditional generative adversarial networks for convolutional face generation. *Technical Report*, page 3, 2015.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.

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.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. DRAW: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pages 1462–1471. PMLR, 2015.

Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

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.

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

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Transactions on Graphics*, 35(4):110, 2016.

Gabriel Ilharco, Yuan Zhang, and Jason Baldridge. Large-scale representation learning from visually grounded untranscribed speech. *arXiv preprint arXiv:1909.08782*, 2019.

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.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017.

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

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

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.

Minguk Kang and Jaesik Park. ContraGAN: Contrastive learning for conditional image generation. *Advances in Neural Information Processing Systems*, 33:21357–21369, 2020.

Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022.

Gwanghyun Kim and Jong Chul Ye. Diffusionclip: Text-guided image manipulation using diffusion models. 2021.

Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: Video story QA by deep embedded memory networks. *arXiv preprint arXiv:1707.00836*, 2017.

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

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pages 3581–3589, 2014.

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

Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, and Marc'Aurelio Ranzato. Fader networks: Manipulating images by sliding attributes. In *Advances in Neural Information Processing Systems*, pages 5967–5976, 2017.

Kwot Sin Lee, Ngoc-Trung Tran, and Ngai-Man Cheung. Infomax-GAN: Improved adversarial image generation via information maximization and contrastive learning. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 3942–3952, 2021.

Bowen Li and Thomas Lukasiewicz. Learning to model multimodal semantic alignment for story visualization. In *Findings of EMNLP 2022*, December 2022a.

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

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

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

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

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, 2020c.

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, 2022a.

Bowen Li, Philip HS Torr, and Thomas Lukasiewicz. Memory-driven text-to-image generation. *arXiv preprint arXiv:2208.07022*, 2022b.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020d.

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, 2019b.

Yijun Li, Lu Jiang, and Ming-Hsuan Yang. Controllable and progressive image extrapolation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 2140–2149, 2021.

Yitong Li, Zhe Gan, Yelong Shen, Jingjing Liu, Yu Cheng, Yuexin Wu, Lawrence Carin, David Carlson, and Jianfeng Gao. StoryGAN: A sequential conditional GAN for story visualization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6329–6338, 2019c.

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.

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 *European Conference on Computer Vision*, pages 740–755. Springer, 2014.

Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

Adyasha Maharana and Mohit Bansal. Integrating visuospatial, linguistic and commonsense structure into story visualization. *arXiv preprint arXiv:2110.10834*, 2021.

Adyasha Maharana, Darryl Hannan, and Mohit Bansal. Improving generation and evaluation of visual stories via semantic consistency. *arXiv preprint arXiv:2105.10026*, 2021.

Adyasha Maharana, Darryl Hannan, and Mohit Bansal. Storydall-e: Adapting pretrained text-to-image transformers for story continuation. In *European Conference on Computer Vision*, pages 70–87. Springer, 2022.

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*, pages 25–32. IEEE, 2020.

Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*, 2015.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International Conference on Machine Learning*, pages 3481–3490. PMLR, 2018.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

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

Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 299–307, 2017.

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.

Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4467–4477, 2017.

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.

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.

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.

Aude Oliva, Antonio Torralba, Monica S Castelhano, and John M Henderson. Top-down control of visual attention in object detection. In *Proceedings 2003 International Conference on Image Processing*, volume 1, pages 253–256. IEEE, 2003.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Xichen Pan, Pengda Qin, Yuhong Li, Hui Xue, and Wenhu Chen. Synthesizing coherent story with auto-regressive latent diffusion models. *arXiv preprint arXiv:2211.10950*, 2022.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019.

Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, pages 319–345. Springer, 2020.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

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.

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, 2019a.

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, 2019b.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

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 *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

Tanzila Rahman, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, Shweta Mahajan, and Leonid Sigal. Make-a-story: Visual memory conditioned consistent story generation. *arXiv preprint arXiv:2211.13319*, 2022.

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.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

Scott Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. In *Advances in Neural Information Processing Systems*, pages 217–225, 2016a.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, 2016b.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

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.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

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.

Amaia Salvador, Xavier Giró-i Nieto, Ferran Marqués, and Shin'ichi Satoh. Faster r-cnn features for instance search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–16, 2016.

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. *arXiv preprint arXiv:2301.09515*, 2023.

Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

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

Yun-Zhu Song, Zhi Rui Tam, Hung-Jen Chen, Huiao-Han Lu, and Hong-Han Shuai. Character-preserving coherent story visualization. In *European Conference on Computer Vision*, pages 18–33. Springer, 2020.

Danail Stoyanov, Zeike Taylor, Gustavo Carneiro, Tanveer Syeda-Mahmood, Anne Martel, Lena Maier-Hein, João Manuel RS Tavares, Andrew Bradley, João Paulo Papa, Vasileios Belagiannis, et al. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings*, volume 11045. Springer, 2018.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

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.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.

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.

Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, pages 4790–4798, 2016.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision*, pages 268–285. Springer, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(12), 2010.

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

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015.

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.

Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Clusterfit: Improving generalization of visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6509–6518, 2020.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29, 2016.

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, 2017a.

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, 2018a.

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 Conference on Computer Vision and Pattern Recognition*, pages 833–842, 2021.

Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016.

Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang. Progressive attention guided recurrent network for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 714–722, 2018b.

Zizhao Zhang, Yuanpu Xie, Fuyong Xing, Mason McGough, and Lin Yang. MDNet: A semantically and visually interpretable medical image diagnosis network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6428–6436, 2017b.

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

Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems*, pages 487–495, 2014b.

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.

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.

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.

C Lawrence Zitnick and Devi Parikh. Bringing semantics into focus using visual abstraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3009–3016, 2013.

C Lawrence Zitnick, Devi Parikh, and Lucy Vanderwende. Learning the visual interpretation of sentences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1681–1688, 2013.