Pontifícia Universidade Católica
do Rio de Janeiro

Vinicius Gomes Pereira

Improving text-to-image synthesis with U2C - Transfer Learning

Dissertação de Mestrado

Dissertation presented to the Programa de Pós–graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática.

Advisor : Prof. Eduardo Sany Laber
Co-advisor: Prof. Jônatas Wehrmann

Rio de Janeiro
August 2023

Pontifícia Universidade Católica
do Rio de Janeiro

Vinicius Gomes Pereira

Improving text-to-image synthesis with U2C - Transfer Learning

Dissertation presented to the Programa de Pós–graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática. Approved by the Examination Committee:

Prof. Eduardo Sany Laber
Advisor
Pontifícia Universidade Católica do Rio de Janeiro – PUC-Rio

Prof. Jônatas Wehrmann
Co-advisor
Teia Labs

Prof. Sérgio Colcher
Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio

Prof. Julio Cesar Duarte
Instituto Militar de Engenharia - IME

Rio de Janeiro, August 17th, 2023

Vinicius Gomes Pereira

Graduated in Computer Engineering from Instituto Militar de Engenharia (IME) and Master in Mathematical Modeling of Information from Fundação Getúlio Vargas (FGV).

Bibliographic data

Gomes Pereira, Vinicius

Improving text-to-image synthesis with U2C - Transfer Learning / Vinicius Gomes Pereira; advisor: Eduardo Sany Laber; co-advisor: Jônatas Wehrmann. – 2023.

66 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2023.

Inclui bibliografia

1. Informática – Teses. 2. Síntese de Imagens. 3. Aprendizado Multimodal. 4. Transferência de aprendizado. 5. Redes Generativas Adversárias. I. Laber, Eduardo. II. Wehrmann, Jônatas. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

Ao incessante fluxo e refluxo da vida,
seus ritmos, suas surpresas, suas profundezas insondáveis,
esta dissertação é dedicada.

Acknowledgments

I am deeply grateful to everyone who has supported me while writing this dissertation.

A heartfelt thanks to my family for their constant encouragement and support. To my husband Alberto, your patience, love, and faith in me have been crucial in this journey. Your sacrifices and encouraging words have meant a lot. I must also mention my loyal puppy Panqueca, who has brought joy and companionship to my life during this academic journey.

I'm grateful to my friends, Camila and Matheus, for always being there and providing motivation. I also want to thank my friends from PUC, Daniel and Dayson, for enriching this experience with academic discussions and shared experiences.

I owe a lot to my advisor and friend, Jônatas. His guidance, dedication, and feedback have been instrumental in completing this dissertation. I have learned so much under his mentorship. I also wish to express my gratitude to Eduardo Laber for accepting me as a student, undertaking this task, and offering his constant assistance.

This has been a team effort and I've been fortunate to have an incredible support system. Thanks to all for your contributions to my growth.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Gomes Pereira, Vinicius; Laber, Eduardo (Advisor); Wehrmann, Jônatas (Co-Advisor). **Improving text-to-image synthesis with U2C - Transfer Learning**. Rio de Janeiro, 2023. 66p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Generative Adversarial Networks (GANs) are unsupervised models that can learn from an indefinitely large amount of images. On the other hand, models that generate images from language queries depend on high-quality labeled data that is scarce. Transfer learning is a known technique that alleviates the need for labeled data, though it is not trivial to turn an unconditional generative model into a text-conditioned one. This work proposes a simple, yet effective fine-tuning approach, called Unconditional-to-Conditional Transfer Learning (U2C transfer). It can leverage well-established pre-trained models while learning to respect the given textual condition conditions. We evaluate U2C transfer efficiency by fine-tuning StyleGAN2 in two of the most widely used text-to-image data sources, generating the Text-Conditioned StyleGAN2 (TC-StyleGAN2). Our models quickly achieved state-of-the-art results in the CUB-200 and Oxford-102 datasets, with FID values of 7.49 and 9.47, respectively. These values represent relative gains of 7% and 68% compared to prior work. We show that our method is capable of learning fine-grained details from text queries while producing photorealistic and detailed images. Our findings highlight that the images created using our proposed technique are credible and display a robust alignment with their corresponding textual descriptions.

Keywords

Image Synthesis; Multimodal Learning; Transfer Learning; Generative Adversarial Networks.

Resumo

Gomes Pereira, Vinicius; Laber, Eduardo; Wehrmann, Jônatas. **Aprimorando a síntese de imagens a partir de texto utilizando transferência de aprendizado U2C**. Rio de Janeiro, 2023. 66p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

As Redes Generativas Adversariais (GANs) são modelos não supervisionados capazes de aprender a partir de um número indefinidamente grande de imagens. Entretanto, modelos que geram imagens a partir de linguagem dependem de dados rotulados de alta qualidade, que são escassos. A transferência de aprendizado é uma técnica conhecida que alivia a necessidade de dados rotulados, embora transformar um modelo gerativo incondicional em um modelo condicionado a texto não seja uma tarefa trivial. Este trabalho propõe uma abordagem de ajuste simples, porém eficaz, chamada U2C transfer. Esta abordagem é capaz de aproveitar modelos pré-treinados não condicionados enquanto aprende a respeitar as condições textuais fornecidas. Avaliamos a eficiência do U2C transfer ao ajustar o StyleGAN2 em duas das fontes de dados mais utilizadas para a geração images a partir de texto, resultando na arquitetura Text-Conditioned StyleGAN2 (TC-StyleGAN2). Nossos modelos alcançaram rapidamente o estado da arte nas bases de dados CUB-200 e Oxford-102, com valores de FID de 7.49 e 9.47, respectivamente. Esses valores representam ganhos relativos de 7% e 68%, respectivamente, em comparação com trabalhos anteriores. Demonstramos que nosso método é capaz de aprender detalhes refinados a partir de consultas de texto, produzindo imagens fotorrealistas e detalhadas. Além disso, mostramos que os modelos organizam o espaço intermediário de maneira semanticamente significativa. Nossas descobertas revelam que as imagens sintetizadas usando nossa técnica proposta não são apenas críveis, mas também exibem forte alinhamento com suas descrições textuais correspondentes. De fato, os escores de alinhamento textual alcançados por nosso método são impressionantemente e comparáveis aos das imagens reais.

Palavras-chave

Síntese de Imagens; Aprendizado Multimodal; Transferência de aprendizado; Redes Generativas Adversárias.

Table of contents

List of figures

List of tables

List of Abreviations

GAN – Generative Adversarial Network

U2C – Unconditional to Conditional

FID – Fréchet Inception Distance

KID – Kernel Inception Distance

IS – Inception Score

SN – Spectral Normalization

MRR – Mean Reciprocal Rank

ADA – Adaptive Discriminator Augmentation

DAMSM – Deep Attentional Multimodal Similarity Model

KL Divergence – Kullback–Leibler divergence

TC-StyleGAN – Text-Conditioned StyleGAN

NLP – Natural Language Processing

CLIP – Contrastive Language-Image Pre-Training

T2I – Text to Image

RNN - Recurrent Neural Network

CNN - Convolutional Neural Network

LSTM - Long Short-Term Memory

ViT - Vision Tranformer

U2C transfer - Unconditional-to-Conditional Transfer Learning

Life can only be understood backwards; but it must be lived forwards

Søren Kierkegaard, *Journalen JJ:167 (1843).*

1
Introduction

Generating realistic images from human-written sentences is a challenging research area. In recent years, many novel deep network frameworks to deal with this task have successfully been implemented, and these architectures are also rapidly evolving, which increases the potential development of applications to handle real-world problems in the area of image editing, visual effects, and the design industry. To address the problem of artificial image synthesis, Generative Adversarial Networks (GANs) (5, 6) have emerged as architecture with promising results (7, 8), and recently, diffusion models (9, 10) and autoregressive models (11), as generative frameworks that synthesizes images with high variability and trustworthiness.

Generation of images based on detailed natural language descriptions is an even more difficult task, as it inserts the representation of natural language into the problem. It is noteworthy that albeit unconditional GANs are able to learn from an indefinitely large amount of images, text-to-image models depend on high-quality labeled data that is scarce. Transfer learning (12) is a known technique that alleviates the need for labeled data, though it is not trivial to turn an unconditional generative model into a text-conditioned one.

We introduce Unconditional-to-Conditional Transfer Learning (U2C transfer), which allows the use of pre-training information from an unconditional network to generate text-conditioned images. It is able to leverage pre-trained models that generate Human Faces (13) and adapt them to synthesize high-quality images of Birds (14) and Flowers (15), for instance. Such images do present fine-grained details that respect the input textual query.

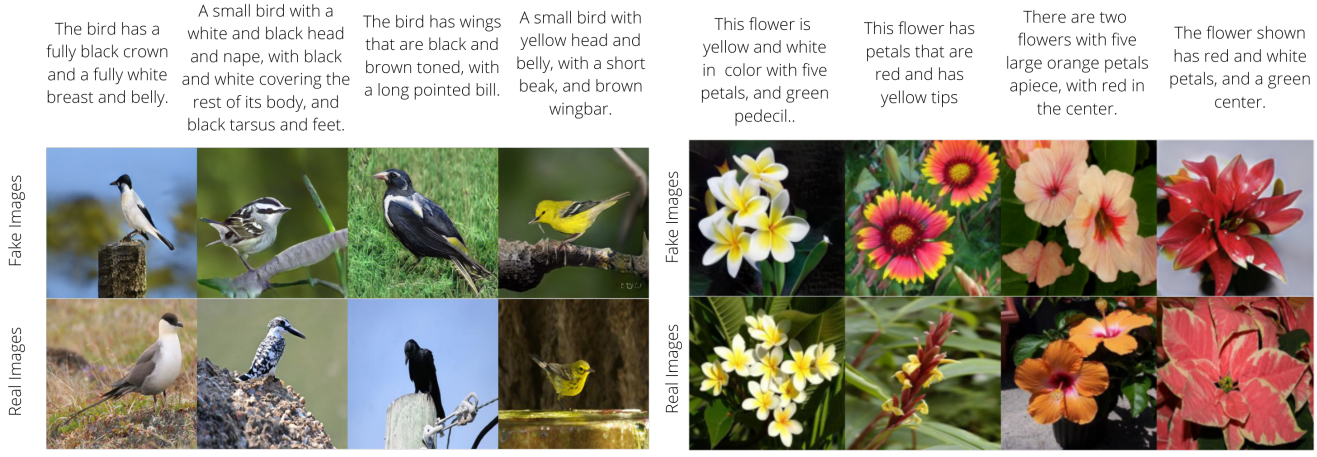

Figure 1.1: Images synthesized by the proposed approach and the ground truth

This method not only stabilizes the training process but also makes the training convergence faster. In addition, note that it is challenging to train a text-conditioned model on these datasets given that they present a rather limited amount of images, which the discriminator easily overfits (16). It is also complex to get proper textual representations that contain rich details regarding the data distribution of the dataset. Moreover, with few examples of captions, the space of the sentence representation is highly discontinuous, highlighting the need for a better design for such text-encoder.

U2C transfer is tested in the standard StyleGAN2 (7) unconditional models by finetuning them in two widely used datasets. The resulting model, a Text-Conditioned StyleGAN2 (TC-StyleGAN2 for short), effortlessly bests prior work results in a few hundred iterations. We try to follow and reuse the maximum number of pre-trained weights as possible, making only indispensable changes to the architecture. TC-StyleGAN2 also makes use of two data augmentation mechanisms to help preventing overfitting: (i) a textual augmentation technique (17), to increase smoothness and continuity of the conditional text space; and (ii) adaptive discriminator augmentation (ADA) (18) which automatically regulates the strength of the image transformations. We chose to adopt StyleGAN2's due to its image quality results, as well as the implementation of a mapping network module, which generates an intermediate and less entangled latent representation. A less entangled representation of noise and textual representation is critical for the network to produce reliable results.

We quantitatively evaluate our models in terms of Fréchet Inception Distance (FID), Kernel Inception Distance (KID), Inception Score (IS), and text-alignment metrics. TC-StyleGAN2 achieved FID of 7.49 and 9.47 for CUB-200 and Oxford-102 Flowers data, respectively. We also evaluate qualitatively by demonstrating that this model learns regular structures that allow image editing via vector arithmetic operations in both condition and intermediate latent spaces.

The motivation for using the pre-trained models is that there are many more unconditional models than conditional ones, given that they are easier to train, have less complexity, and can leverage from more data. By using more data, one can have higher-quality pre-trained weights. We propose an approach that allows using such high-quality models in datasets that contain very limited amounts of data.

In summary, we highlight the following contributions:

– We introduce a new transfer learning adaptation technique that involves modifications to inputs, outputs, and loss function, using an uncondi-

tional model. Our method has several advantages: it is easy to implement, trains faster, and achieves state-of-the-art results in a few iterations in two widely used datasets. Therefore, by using U2C Transfer, potentially all unconditional models can be used for training text-conditional models. To the best of our knowledge, this is the first transfer-learning approach that allows that. The specific modifications we proposed were designed carefully to allow reusing the maximum amount of weights, minimize complexity and avoid training collapse.

- We demonstrate that TC-StyleGAN2 enables image editing using natural language through arithmetic operations not only in the conditional space but also in the intermediate mapping space $\mathcal{W}$, being able to modify several aspects of the image like colors, sizes, and some specific details. We showed that $\mathcal{W}$ space carries a meaningful and learned sentence representation.
- Our approach effectively addresses several challenges, including text-space discontinuity, training stabilization, weights initialization, and slow convergence time. By incorporating these key elements, our proposed method offers a holistic solution for enhancing the training process of text-to-image models in scenarios where data availability is restricted.
- We proposed a new metric for assessing the level of correspondence between textual and visual inputs. This metric represents a novel approach to evaluating the quality and accuracy of synthesized images concerning their textual descriptions.

The structure of this document is as follows: Chapter 2 reviews the background concepts of Generative Adversarial Networks, including a discussion of the challenges involved in training generative models and an overview of related studies. In Chapter 3, our method, the Unconditional to Conditional Transfer Learning, is introduced. Chapter 4 presents our main results and compares them with other relevant studies. Finally, in Chapter 5, we discuss the conclusions drawn from our findings and the challenges that must be addressed.

2
Background and Related Work

In this chapter, we will discuss the syntheses of artificial images by employing generative adversarial networks (GANs). We will examine the evolution of GAN architecture and training techniques for generating images based on text. We will address the challenges that arise when training and evaluating models regarding quality, diversity, and textual alignment. We will also introduce the transfer learning concept and conclude the section with related works on transfer learning in GANs.

2.1
Generative Adversarial Networks

GANs were introduced by Goodfellow et al.(5) in 2014, and are robust models capable of learning complex distributions to generate samples with semantic meaning. GANs employ two neural networks, a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$. In unconditional GANs, $\mathcal{G}(\mathbf{z}, \theta_g)$ is parameterized by θ_g, and it receives a random noise $\mathbf{z}$, sampled from a noise space $\mathcal{Z}$ as input, producing an output distribution p_g, that approximates the real data distribution p_{data} (5). The discriminator $\mathcal{D}(\mathbf{x}, \theta_d)$ outputs a single scalar, that represents a probability of the input $\mathbf{x}$ belongs to the distribution p_{data} rather than the generator distribution p_g. During the training process, both the generator and the discriminator are trained together in a competitive manner, which involves playing a min-max game. The generator's objective is to produce synthetic images that deceive the discriminator, whereas the discriminator's aim is to differentiate between the artificial and authentic images, minimizing the final classification error. The generative neural network is trained to maximize the final classification error. The value function $V(\mathcal{G}, \mathcal{D})$ is described in Equation 2-1:

$$\min_{\mathcal{G}} \max_{\mathcal{D}} V(\mathcal{G}, \mathcal{D}) = \mathbb{E}_{\mathbf{x} \sim p_{data}}[log(\mathcal{D}(\mathbf{x}, \theta_d))] + \mathbb{E}_{\mathbf{z} \sim p_Z}[log(1 - \mathcal{D}(\mathcal{G}(\mathbf{z}, \theta_g))] \quad (2\text{-}1)$$

Hence, $\mathcal{D}$ will try to maximize its output when the input comes from a real distribution and minimize it when it was generated from $\mathcal{G}$. Initially, $\mathcal{G}$ is not producing data similar to the real ones, and the discriminator $\mathcal{D}$ easily rejects the synthetic data, saturating the second term of Equation 2-1, producing weak gradients to the generator $\mathcal{G}$ (5). That's why, Equation 2-1 is rewritten, so that $\mathcal{G}$ is trained to maximize and $\mathcal{D}$ to minimize a modified and equivalent value function, as described in Equation 2-2:

$$\max_{\mathcal{G}} \min_{\mathcal{D}} V(\mathcal{G}, \mathcal{D}) = \mathbb{E}_{\mathbf{x} \sim p_{data}}[-log(\mathcal{D}(\mathbf{x}, \theta_d))] + \mathbb{E}_{\mathbf{z} \sim p_Z}[log(\mathcal{D}(\mathcal{G}(\mathbf{z}, \theta_g))] \quad (2\text{-}2)$$

Training GANs can be challenging due to certain instabilities, and Wiatrak, Albrecht e Nystrom(19) have identified three challenges that arise during the training process:

- Mode Collapse: this happens when the generator produces samples with low diversity, failing to generate realistic and diverse outputs, mapping multiple distinct inputs to the same output.
- Convergence/Training Stability: training process can be unstable, resulting in oscillations, convergence problems, and cyclical behavior. The existence of a global Nash Equilibrium in training has been proven in the study (5), but reaching it is not simple.
- Vanishing gradients: as the discriminator improves its ability to differentiate between real and generated samples, the updates to the generator consistently degrade, resulting in almost zero gradient values.

Subsequent advancements towards generating more reliable, higher-resolution, and diverse data depend on the progress of techniques to mitigate the previously mentioned issues: mode collapse, convergence and training stability, and vanishing gradients. Section 2.1.1 will provide an overview of recent advancements in these areas.

2.1.1 Improvements in Training Stabilization

In the standard implementation of GANs, proposed by Goodfellow et al.(5), the authors have demonstrated that the value function employed, described in Equation 2-1, leads to the optimization of Jensen-Shannon divergence (JSD) between the real and fake distribution. A divergence function measures the similarity between two probability distributions, and JSD belongs to the f-divergence family. Following the original work on GANs, researchers (20, 21) have explored alternative value functions that also optimize the f-divergence family to enhance and stabilize the training process. In particular, Mao et al.(21) introduced the Least Squares GAN (LSGAN), which utilizes the conventional GAN implementation with a least squares metric as the value function. LSGAN's training leads to the optimization of the Pearson χ^2-divergence. The LSGAN study demonstrated that they overcame the vanishing gradient issue by implementing these modifications.

The study conducted by Arjovsky, Chintala e Bottou(22), which led to the development of the WGAN architecture, investigated the application of

value functions that are not part of the f-divergence family. Instead, they used the Wasserstein distance, also called the Earth Mover's distance. The Wasserstein distance quantifies the minimum amount of effort needed to transform one probability distribution into another (22). In this study, the output of the discriminator is no longer binary but rather a real-valued score, also called as critic. The Wasserstein distance significantly improves the problem of training stability and convergence problem, also showing no evidence of mode-collapse in the experiments conducted. Since the discriminator outputs a real number, the clipping of values above a threshold is applied, enforcing a Lipschitz constraint on the critic. Zhou et al.(23) demonstrated that the clipping procedure is necessary. They showed that if clipping is not applied, the training process may suffer from a weak gradient signal produced by the discriminator.

Gulrajani et al.(24) improved WGAN and argued that although there have been advancements in the training stability of WGAN, the use of weight clipping introduces an unstable behavior in training, taking more time to converge. Thus, WGAN-GP is introduced, which applies a gradient penalty in the critic loss function instead of using clipping but also ensures a 1-Lipschitz constraint on the value function.

Applying normalization to the discriminator in GANs can also have a stabilizing effect on the training process. It can lead to more stable optimization by encouraging the discriminator to provide higher-quality feedback and enhance the representation of the network's layers within the discriminator (19).

Ioffe e Szegedy(25) introduced the batch normalization technique, which helps address the issue known as internal covariate shift. This problem arises when the distribution of each layer changes due to the modification of previous layers, resulting in training instabilities. In order to overcome this issue, lower learning rates and carefully selected hyperparameters are used, which can make training more challenging. The authors proposed the normalization of the activations of each layer using a mini-batch, incorporating it into the model architecture. This technique improved training stabilization and the vanishing gradient issue and allowed a more freedom exploration of higher learning rates and a wider range of hyperparameters. It also acts as a regularizer, reduces the training time, and eliminates the need for dropout in some cases (25).

Batch normalization is sensitive to batch size and may not be applicable to recurrent neural networks. Layer normalization was introduced by Ba, Kiros e Hinton(26) to overcome these limitations. This technique performs the normalization computation the same way during training and testing. Unlike batch normalization, layer normalization does not introduce dependencies or

bias between training cases, as it applies the normalization procedure using only a single training example. It computes the mean and variance from all of the summed inputs to the neurons in a layer. As a result, layer normalization also overcomes the interval covariate shift issue, and it has been shown to speed up the training of neural networks and to be highly effective in stabilizing the hidden state dynamics in recurrent networks (26).

Similarly, weight normalization (27) is a technique inspired by batch normalization and also is applied to one training example. However, weight normalization applies normalization to the weights of the neural network. It is shown that weight normalization also speeds up training and increases stability.

Also addressing the training stabilization problem using normalization procedures, Miyato et al.(28) introduced a new technique called spectral normalization (SN) to stabilize the discriminator training process, being computationally efficient and easily integrated into existing implementations. SN is a weight normalization technique that restricts the spectral norm of each weight matrix to a fixed value. The spectral norm is the largest singular value of a matrix, which provides an upper bound on the Lipschitz constant. Through this constraint mechanism, SN aids in preventing sudden fluctuations in network weights during training, which can lead to instability and hinder the model's generalization performance.

Salimans et al.(29) introduced several architectural features and training procedures to tackle issues that arise in training GANs. The proposed techniques included feature matching, mini-batch discrimination, historical averaging, and virtual batch normalization. Specifically, mini-batch discrimination was shown to be particularly effective in addressing mode-collapse, while virtual batch normalization and historical averaging were found to improve optimization and convergence during training.

As the previously discussed techniques continued to evolve, so did the architectures used in generative models, enabling the synthesis of high-resolution, diverse, and realistic images. The forthcoming section, Section 2.1.2, will provide a more comprehensive exploration of these developments.

2.1.2 Improvements in Architecture

Since their debut in 2014, several changes have been made to the architecture of GANs. One significant development is DCGAN (Deep Convolutional Generative Adversarial Networks) (30), motivated by the effectiveness of CNNs in supervised image classification tasks. DCGAN's architecture consists entirely of convolutional layers, replacing pooling layers in the discriminator and

the generator with different types of convolution layers. Additionally, batch normalization is applied after each convolutional layer to enhance the gradient flow and stability.

It is evident that the discriminator overpowering the generator in the early stages of the game has an impact on GANs training (19). Generating high-resolution images is a matter that requires changes in the network architecture.

Denton et al.(31) proposed LAPGAN, employing a series of convolutional networks arranged in a Laplacian pyramid framework to create images in a step-by-step manner, starting from rough to fine details. The image generation process is split into several stages of refinement. At each level of the pyramid, a distinct generative CNN model is trained using the standard GAN training method (5). To address the challenge of generating high-quality images with fine-grained details, the researchers behind the Progressive GAN (PROGAN) (32) proposed a method that trains the generator and discriminator networks in multiple phases. This approach, known as progressive growing, gradually increases the image resolution and complexity in each phase by adding layers to the architectures of both networks. The researchers also introduced several strategies to improve training stability and prevent issues like mode-collapse, including pixel-wise feature vector normalization and mini-batch standard deviation. Overall, the progressive growing method has proven to be a powerful strategy for generating high-resolution images with impressive visual fidelity.

In 2017, Vaswani et al.(33) proposed transformers, which are a type of neural network based on attention mechanisms. Since their introduction, transformers have become the state-of-the-art technique for many NLP tasks. They excel at modeling long-range dependencies and capturing both local and global connections. In (34), it is introduced SAGAN (Self-Attention Generative Adversarial Networks), using a self-attention mechanism that complements convolutional layers in order to capture long-range dependencies in pixels. When the study was published, SAGAN achieved state-of-the-art results and has since proven effective for generating high-quality images.

Inspired by SAGAN, BigGAN (8) is a generative model that generates high-quality images across a wide range of categories. It uses several key features, such as the self-attention module to capture fine-grained details, the hinge loss function to prevent mode collapse, and spectral normalization to stabilize training. Additionally, it employs the truncation trick to control the degree of variation in generated images and a moving average of the model weights during inference to stabilize the output. Other important features include orthogonal weight initialization and larger batch sizes for more

efficient use of parallel hardware. Overall, BigGAN is a powerful and versatile generative model that has achieved impressive results, generating images with 512x512 pixels of resolution. BigGAN has achieved the most impressive results among GAN-based models for ImageNet (35).

Another category of methods, known as style transfer techniques (36, 37, 38), has become increasingly popular due to their impressive results. Inspired by these techniques, the StyleGAN architecture (39) has had a significant impact on image synthesis. The focus of its development has been on the generator side, utilizing the Adaptive Instance Normalization (AdaIn) (36) to perform style transfer. StyleGAN introduced the Mapping Network, which maps input latent noise into an intermediate space. This network aims to learn a less entangled intermediate space, as an entangled space may cause undesired changes in the image during noise vector interpolation, as previously noted by (40). Two metrics were proposed to measure the entanglement of the space: the perceptual path length, which measures the degree of change in the image during interpolation, and the linear separability, which evaluates how well latent-space points can be divided into two separate sets that correspond to a specific binary attribute of the generated image, using a linear hyperplane. Analysis of these metrics has shown that implementing the mapping network is beneficial for achieving a less entangled generator and also improving image quality scores. Notably, models from the StyleGAN family have demonstrated impressive image synthesis capabilities for single-category domains on datasets such as (14, 15, 13, 41).

StyleGAN has continued to evolve with subsequent versions like StyleGAN2 and StyleGAN3, each introducing improvements to the original architecture. The authors identified and fixed several image quality issues in StyleGAN. StyleGAN2 introduced a path length regularization technique to encourage the generator to produce smoother and more continuous output images. They also proposed an alternative design of progressive growing techniques (32) for generating high-resolution images. StyleGAN2 is designed to be more computationally efficient, allowing it to generate high-resolution images such as 1024x1024 pixels faster than its predecessor. StyleGAN3 (42) made small architectural changes that do not impact quality metrics such as FID (43) but prevent texture sticking on the generated images, for example.

Recent models such as (9, 11, 10, 44), have marked a new era of image generation, showcasing unprecedented photorealism and diversity in the generated images. The development of diffusion models (45, 46), autoregressive transformers (11), and large-scale language encoders (47, 48) has enabled text-to-image synthesis to achieve an unmatched level of diversity and realism never

achieved using GANs techniques.

StyleGAN exhibits significant degradation in performance when dealing with large unstructured datasets such as ImageNet (49). However, it excels in specific domains, such as (13, 41, 14, 15). Compared to autoregressive and diffusion models, GANs have a notable advantage in terms of speed and smooth latent space, allowing for more controllable synthesis. However, their training can be unstable, and the diversity of generation may be limited, as highlighted in (50).

Trying to expand GAN's capabilities to more diverse datasets is a challenge that is being investigated. This limitation is researched and addressed in (49, 50, 51). In their study, Sauer et al.(52) introduced a training strategy that permits the training of other GANs like StyleGAN to be more stable, using pre-training information. By adopting the Projected GAN paradigm and proposing modifications, Sauer, Schwarz e Geiger(49) developed StyleGAN-XL, which successfully trains the latest StyleGAN3 generator on ImageNet. The resulting model generates high-quality images of 1024x1024 resolution, achieving state-of-the-art results in image quality metrics at this resolution.

In the field of unconditional image generation using generative adversarial networks, the current focus is on expanding the diversity of images generated with high resolution without being restricted to specific classes.

In section 2.2, we will deal with advances in the area of generating text-conditioned images.

2.2 Text-to-image Synthesis

Synthesizing images based on comprehensive natural language descriptions is more challenging, requiring incorporating natural language representation into the problem. Text-to-image models rely on high-quality labeled data that is often limited in availability.

This section will focus on text-to-image tasks using GANs. Specifically, we will examine text encoders in Section 2.2.1 and architectural advancements in Section 2.2.2.

2.2.1 Text Encoders

Text encoders play a crucial role in image generation as they enable the use of textual information as a conditioning variable for the generation of images. In image generation tasks, a text encoder takes a textual input, such as a caption or a sentence, and converts it into a numerical representation.

This embedding is then used as a conditioning variable by the image generator to synthesize an image that corresponds to the input text.

In their work, Reed et al.(53) introduced a Deep Symmetric Structured Joint Embedding (DS-SJE) model as a multi-modal approach for linking visual descriptions and images. The model is designed to learn a compatibility function that maximizes the agreement between a given textual description and its corresponding image while minimizing it with images from other categories. The text encoder component of the model is composed of a convolutional neural network (CNN) that operates at the character level, followed by a long short-term memory (LSTM) network. The final model is known as Char-CNN-RNN encoder, and Bag-of-Words and Word2Vec (54) methods were also evaluated and less effective compared to their proposed text encoder model. Char-CNN-RNN text encoder model was used in several architectures like (6, 17, 2).

The space of text embeddings is both highly dimensional and sparse. To increase the smoothness in this space, (6) proposed a sentence interpolation strategy, which involves generating additional text representations. Another solution proposed by (2) is a conditional augmentation mechanism that generates more captions for a given image by sampling from a Gaussian distribution. The mean and covariance of the textual-embedding distribution of the captions for a particular image are used to determine the parameters of the Gaussian distribution. In their work, Souza, Wehrmann e Ruiz(4) also introduced an interpolation approach that involves utilizing all captions associated with a particular image and randomly sampling weights.

Incorporating the attention mechanism into its architecture pipeline, AttnGAN (55) utilizes the Deep Attentional Multimodal Similarity Model (DAMSM) to measure the similarity between an input image and text description. The DAMSM model consists of a text encoder and an image encoder, which are trained to extract word and sentence embeddings and meaningful image features, respectively. The text encoder is implemented using a bi-directional LSTM network, while the image-encoder is built upon the Inception-v3 (56) model. The model incorporates an attention mechanism to produce attention maps identifying the most relevant words for each image sub-region. The attention-driven image-text matching score is then computed in the same multimodal space, ensuring that the image representation accurately reflects the intended text description. Ye et al.(57) improved AttnGan (55) and DM-GAN (58), in their work, using a contrastive learning approach. In the pre-training stage, this technique was employed in the DAMSM module, learning a more consistent textual representation. Then, this method was utilized in the GANs training, enhancing the consistency between the gener-

ated images and their respective captions. Experimental results have shown that the quality of synthesized images has improved significantly in terms of quality metrics.

Also training in a multimodal paradigm, CLIP (Contrastive Language-Image Pre-Training) (48) is a neural network model that was trained on a large corpus of text and images from the internet, with over 400 million images and captions paired. CLIP is trained using a contrastive loss function that maximizes the similarity between corresponding image and text representations while minimizing the similarity between mismatched pairs. Using state-of-art approaches for the image and text encoder, CLIP employed text transformers (59) as text encoder and vision transformers (60) as image encoders. The resulting model achieves state-of-the-art performance on a range of benchmark datasets, demonstrating the approach's effectiveness.

In Section 2.2.2, we will provide an overview of a text-conditioned architecture.

2.2.2 Improvements in the Text-Conditioned Architecture

Reed et al.(6) proposed the first end-to-end differentiable architecture, GAN-INT-CLS, to generate images from a text that operates at both character and pixel levels. The architecture is built upon a DC-GAN and utilizes char-CNN-RNN (53) to encode the input sentence. This encoded representation is then concatenated with noise and used as input to the network. During training, the discriminator is trained to distinguish between an authentic picture and its corresponding text, a real picture with an incorrect caption, and a fake picture with its related text. This architecture can only generate images with small resolutions of 64x64 pixels. The following works proposed to use multiple stacked generators to generate higher resolutions. Subsequent works presented the use of multiple stacked generators to create higher-resolution images.

Zhang et al.(17) introduced Stacked Generative Adversarial Networks (StackGAN). StackGAN can generate images of resolution 256×256 pixels, conditioned to text, in multiple stages. In the first stage, low-resolution images are generated, outlining the basic features of the object described in the text. The Stage-I results are then input to Stage II along with the original text descriptions to produce high-resolution images with realistic details.

StackGAN++ (2) was proposed as an updated version of StackGAN and also employs multiple stages for image generation. In contrast to the original StackGAN, StackGAN++ incorporates a joint conditional and unconditional

distribution approximation in the value function. This enables the model to approximate both conditional and unconditional image distributions by calculating the unconditional error of the generated image in addition to the conditional error, which is conditioned on the text embedding. Furthermore, a color regularization technique was also implemented. The study demonstrated that these improvements effectively prevent mode collapse during training.

Using a single generator, HDGAN (3) is a generative model designed to produce high-resolution images with realistic details, implemented using progressive growing architecture. The progressive growing technique enables the model to gradually increase the resolution of generated images, resulting in higher-fidelity details.

Incorporating an attention mechanism to its architecture pipeline, AttnGAN (55) was built upon StackGan++. AttnGAN utilizes the Deep Attentional Multimodal Similarity Model (DAMSM) to measure the similarity between an input image and text description. The generator uses a hierarchical architecture to progressively generate the image in multiple stages. The attention mechanism in AttnGAN operates at multiple levels, at the word and at the sentence level, allowing the generator to focus on relevant information and generate images that closely match the text description.

Some other architectures in text-to-image synthesis adapt unconditional models like (4, 61, 62, 63). BridgeGan, for example, modifies PROGAN architecture, while the study by Souza, Wehrmann e Ruiz(4) implements BigGAN architecture in a text-image task.

Unlike the conventional methods for training text-to-image models, LAFITE (64) is a groundbreaking approach to training text-to-image generation models without any text data. The conventional method of training such models requires a vast number of high-quality image-text pairs, which can be challenging to obtain. However, LAFITE overcomes this obstacle by employing CLIP. LAFITE utilizes the image embedding extracted from the CLIP image encoder as a substitute for the text embedding, as CLIP was trained to maximize the agreement between these representations. Hence, the method eliminates the need for image and text pairs. LAFITE is designed based on the StyleGAN2 network. The proposed method, a language-free model, has shown better performance than most existing models trained with full image-text pairs.

To match the high diversity and quality of diffusion models in generating images from text, Kang et al.(51) proposed GigaGAN by scaling up StyleGAN2 to increase model capacity and overcome the issue of limited image diversity. However, simply expanding the architecture of StyleGAN2 resulted

in unstable training (51). The authors identified key issues to tackle this challenge and devised novel techniques to enhance model capacity while stabilizing training. They also drew inspiration from methods commonly used in diffusion models and incorporated them into GigaGAN. The authors successfully trained GigaGAN, a one-billion-parameter GAN, on large-scale datasets such as LAION2B-en (65), achieving stability in training. GigaGAN has 1 billion parameters, which is lower than the parameter count of the most significant recent synthesis models, such as Imagen (3.0B), DALL-E 2 (5.5B), and Parti (20B), paving the way for further advancements in GANs models.

Section 2.3 will outline the metrics we use to evaluate these models.

2.3 Evaluating Image Synthesis

We will discuss several metrics commonly used to evaluate the diversity of generated images and their alignment with text. For image quality and diversity, some commonly used are Inception Score, Fréchet Inception Distance, and Kernel Inception Distance. Additionally, metrics such as R-precision, Visual-Semantic Similarity, and Semantic Object Accuracy assess the alignment between images and text. We also consider metrics from image captioning, such as BLEU, METEOR, and CIDr. In the next subsections, we will provide a detailed explanation of each of these metrics.

2.3.1 Inception Score

The Inception Score (IS) is a widely-used metric for evaluating the quality and diversity of generated samples in GAN models. It was introduced in a paper by Salimans et al.(66) and relies on a pre-trained neural network called the Inception Net (56), which was trained on the ImageNet dataset. IS measures both objectivity and variety, so the higher the score, the better.

The average KL divergence between the conditional label distribution $p(y|x)$ of generated samples and the marginal distribution $p(y)$ obtained from all the samples is measured to calculate the Inception Score. If the conditional label distribution has low entropy, it means that the network can produce images of good quality. On the other hand, if the marginal distribution has high entropy, it indicates that the network can generate diverse images.

The Inception Score has been shown to have limitations that need to be considered, according to (67). The IS tends to favor overfitted GANs that memorize all training data, which makes it unable to detect overfitting. Furthermore, since the metric is based only on the Inceptionv3 embeddings

and not the real distribution of images, it may introduce bias from ImageNet, which has many object classes. This bias may cause the metric to prefer models that generate good objects instead of realistic images. Lastly, the IS is an asymmetric measure and also may be affected by image resolution.

IS is defined in Equation 2-3:

$$IS(\mathcal{G}) = exp(\mathbb{E}_{x \sim p_{data}}[D_{KL}(p|x)||p(y)]) \tag{2-3}$$

where $D_{KL}(p|q)$ is the KL-divergence between the distributions p and q, and the exponentiation in this formula is to make easier to compare the values (66)

2.3.2 Fréchet Inception Distance

Heusel et al.(43) introduces the Fréchet Inception (FID) distance, which uses statistics from the images in the training data, and evaluates how far the statistics of the fake images are from the real ones. FID uses activation features from the InceptionV3 to extract information from images, assuming these features as a continuous multivariate Gaussian distribution. FID is calculated as the Wasserstein-2 distance between these Gaussian distributions and is represented in Equation 2-4:

$$FID(r, g) = ||\mu_r - \mu_g||_2^2 + Tr(\sigma_r + \sigma_g - 2(\sum_r \sum_g)^{1/2} \tag{2-4}$$

where $(\mu_r, \sum_r)$ and $(\mu_g, \sum_g)$ are the mean and covariance of the real and generate images distributions.

According to Bińkowski et al.(68), FID is sensitive to the size of the dataset and may produce unreliable results when it has a small number of samples. Another limitation is that it assumes the extracted features follow a Gaussian distribution, which may not always be the case (67).

2.3.3 Kernel Inception Distance

The Kernel Inception Distance (KID) (68) has been applied as a more suitable metric to measure the similarity between artificial and real image distribution. In their study, Bińkowski et al.(68) prove that the estimator of FID a biased, leading to different results based on the sample size. KID applied the Maximum Mean Discrepancy (MMD) between the InceptionV3 representation of fake and real images through a kernel function. As with the FID score, the lower the KID value, the better. KID uses the polynomial kernel, defined in Equation 2-5:

$$k(x, y) = (\frac{1}{d} x^T y + 1)^3 \tag{2-5}$$

where d is the dimension. Bińkowski et al.(68) argue that this kernel was chosen to avoid correlations with the objective of MMD GANs as well as to avoid tuning any kernel parameter, being an unbiased estimador.

2.3.4 R-Precision, Mean Reciprocal Rank and Precision@K

R-Precision is a metric commonly used to evaluate the effectiveness of information retrieval systems. In the context of T2I (text-to-image) tasks, Xu et al.(55) proposed using *R*-Precision to evaluate the text alignment of a model that generates artificial images. This model uses an image encoder to encode the generated images and a text encoder to encode texts from the data sample. The text embedding is then retrieved based on a distance metric, usually cosine similarity, to the image embedding. Then, the top R texts from the retrieved set are examined, where R is the total number of texts aligned to the image. The *R*-Precision score is calculated by dividing the number of correctly retrieved texts (r) by the total number of texts aligned to the image (R), resulting in a score (r/R) ranging from 0 to 1. To experiment, Xu et al.(55) sampled 99 mismatching captions and one caption aligned to the given image and used the 1-precision metric ($R = 1$). Then, the $R-$precision is calculated using the average and standard deviation using a large sample of images. One drawback of this metric is that when using the same text and image encoders that were used to train the model, it can cause the model to overfit. As stated by Zhang et al.(69), many generative models exhibit R-precision scores that are considerably higher than repeating the same experiment but retrieving captions from real images.

Analogous to R-precision, the Mean Reciprocal Rank (MRR) for a query response is calculated as the reciprocal of the rank of the first correct answer. MRR is then computed as the average of these reciprocal ranks over a set of queries Q. Offering a supplementary perspective to R-precision, MRR affords an additional layer of understanding in evaluating the efficiency of the retrieval process. Whereas R-precision concentrates on the proportion of correct retrievals, MRR, in contrast, considers the rank of the initial correct retrieval, thereby assessing the system's speed in procuring a correct text.

Precision@k is another metric commonly used in information retrieval. It evaluates the quality of a list of items ranked by relevance by measuring the proportion of correct results among the top k positions. To put this in the context of the retrieval experiment previously mentioned, Precision@k is calculated by the binary outcome indicating the presence or absence of the retrieved caption in the top K positions among the sorted retrieved captions.

This process is then repeated for all the test captions, calculating the binary outcomes' average and standard deviation.

2.3.5 Visual Semantic Similarity

Visual Semantic (VS) similarity is a method developed by the authors of the HDGAN paper (3) to measure the similarity between a generated image and its corresponding caption in a conditional image generation task. The VS similarity is calculated using two separate neural networks to extract features from the generated image and the corresponding text, respectively. These networks are trained to represent the image and text in a shared multimodal space inspired by prior work (70). Once the features have been extracted, we calculate the VS similarity using the cosine similarity between the image and text feature vectors. A higher VS similarity score indicates that the generated image is visually and semantically similar to the corresponding text, indicating a higher quality of the generated image.

2.3.6 Semantic Object Accuracy

In their work, (71) proposed Semantic Object Accuracy (SOA) as a metric to assess the effectiveness of text-to-image generative models. To evaluate the quality of the generated images, the authors utilized a pre-trained object detector, specifically the YOLOv3 network (72) trained on COCO dataset (73), to identify if the objects depicted in the images matched those described in the corresponding captions. The evaluation of the YOLOv3 detector yielded two metrics, SOA-C and SOA-I. SOA-C measures the average number of images per class in which the given object is detected. In contrast, SOA-I measures the average number of images across all classes in which the desired object is detected.

2.3.7 BLEU, METEOR, CIDEr

In their study, Hong et al.(74) investigated the coherence between captions and their corresponding images using a pre-trained neural network that generates captions from images. To quantify the coherence between the textual and visual components of the generated content, the authors employed metrics commonly used in captioning tasks, including BLEU (75), METEOR (76), and CIDEr (77).

2.4 Related Work

The most desirable scenario for machine learning is having plentiful labeled training instances that accurately represent the same distribution as the test data (78). However, collecting sufficient training data can be expensive, time-consuming, or unrealistic. Transfer learning has become a promising machine learning methodology that focuses on transferring knowledge across domains, thereby alleviating the need for labeled data. Transfer learning involves leveraging knowledge gained from performing a related task in a source domain to enhance the understanding and performance of a current task in a target domain (79). This process of adapting a model trained on a source domain to perform well on a target domain with different distributions is defined as domain adaptation (78). Recently, transfer learning methods on deep learning aim to reduce the time and cost of the training process (80).

Some transfer-learning methods in GANs have been used in conditional generation. Wang et al.(81) proposed the initialization of the weights of WGAN-GP (24) pre-trained on a diverse dataset and then finetuned this model on small datasets. The pre-trained initialization rather than the random one improved the model's results, achieving better results on fewer iterations.

Then, Noguchi e Harada(82) studied a method to reduce the number of weights to be trained by focusing only on learnable batch statistics parameters of the hidden layers of a pre-trained generator. Wang et al.(83) introduced a transfer method based on extracting knowledge from multiple pre-trained GANs through a trainable miner network.

Freezing some generator or discriminator layers has been studied in (84, 85). Mo, Cho e Shin(84) proposed to freeze the lower layers of the discriminator and only finetune the upper layers.

Similarly, Zhao, Cong e Carin(85) showed that low-level generator layers and discriminator trained on large-scale datasets could be transferred to facilitate generation in distinct and small target domains. Working on the generator side, (86) introduced a novel approach known as Unbalanced GANs. This method involves pre-training the generator of the GAN using a variational autoencoder (VAE). Specifically, the VAE is trained first, and the weights of the variational decoder are then transferred to the generator. Finally, the GAN is trained using the pre-trained generator, guaranteeing the stable training of the generator.

Karras et al.(18) has proposed a mechanism that stabilizes training with limited data using an adaptive discriminator augmentation procedure (ADA), that we will add to our architecture to also deal with tiny datasets.

Besides, Karras et al.(18) argue that transfer learning often gives better results than from-scratch training. However, it depends on the diversity of the source dataset instead of the similarity between the domains. Hence, diverse datasets can be used as source domains to generate more specific ones. Using such strategies in unconditional GANs showed promising results in limited datasets, often achieving better results than training from scratch. The aforementioned techniques are easily extended to conditional tasks rather than text-conditioned ones.

Several techniques exist that utilize pre-trained models to enhance the performance of GANs. One key advantage of such techniques is that pre-training can be accomplished without requiring adversarial training. Wang et al.(87) proposed a unified transfer learning method, which can be used for various kinds of image synthesis tasks, like text-to-image, audio-to-image, and image-to-image, using style mixing data triplets computed from pre-trained and unconditional style GANs. Then, the style mixing triplets are used in several image synthesis architectures, like SPADE (88) and StarGanv2 (89), distilling the knowledge from the pre-trained teacher GAN. This technique improved image quality results in different conditional image synthesis tasks. Sauer et al.(52) introduced the Projected GAN, a novel approach to improve the quality of images generated by GANs. Their proposed method involves leveraging pre-trained perceptual feature spaces, which significantly enhances image quality, sample efficiency, and convergence speed. The authors utilized feature pyramids to incorporate multi-scale feedback, while multiple discriminators were employed to evaluate different aspects of the generated images. Additionally, random projections were utilized to better leverage deeper layers of the pre-trained network. They showcase the approach using StyleGAN2 and FastGAN (90) as baselines.

Our transfer-learning approach has the advantage of being simpler while incarnating all the weights of non-conditional StyleGAN2 architecture, which notably achieved better results in text-to-image tasks.

3
Method

In this section, we describe our proposed approach, Text-Conditioned StyleGAN2 (TC-StyleGAN2), depicted in Figure 3.1. StyleGAN2 models are still considered state-of-the-art approaches for training unconditional Generative Adversarial Networks, and their results hold strong even when compared to more recent counterparts such as StyleGAN3. The usual recipe for training such kinds of networks involves large amounts of data and huge computing power, especially for training at larger resolutions. Notably, they can only generate images in an unconditional fashion. That is, one cannot ask the model to generate a particular kind of image using either class information or natural language queries. TC-StyleGAN2 aims to give StyleGAN networks the ability to generate images from textual descriptions while leveraging high-quality existing pre-trained models. We introduce a special kind of fine-tuning that we call Unconditional-to-Conditional Transfer Learning, which allows the fine-tuning of unconditional models, making them conditional.

We hypothesize that by reusing pre-trained weights, one can accelerate and stabilize the training convergence and also achieve better results. Such an adaptation is not trivial since the model has to accept an additional vector that dictates the natural language condition. This must be done without causing the collapse of the pre-trained model, which could be caused by randomly initialized layers, for instance. For that reason, we believe that the modifications should be minimal and added in the right places with parsimony. In addition, TC-StyleGAN2 makes use of strategies to prevent overfitting and increase the image space's smoothness. These strategies are twofold: (i) employing Conditional Augmentation (CA) on the text embedding to allow learning a conditional smooth space using a fixed, discrete set of captions, and (ii) using Adaptive Discriminator Augmentation, which can increase image transformations when the model is able to overfit the data. Following this, we

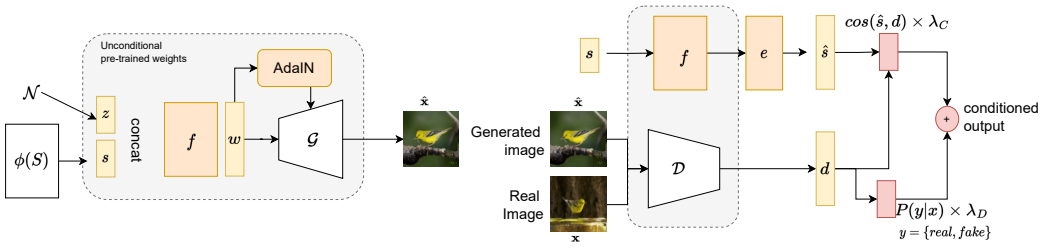

Figure 3.1: Overall architecture of TC-StyleGAN2.

discuss each part of the proposed approach.

3.1 Overall architecture

StyleGAN networks are largely inspired by style-influencing techniques. Those techniques, such as AdaIN, allow the introduction of the input noise vector across multiple stages of the network, allowing it to control the content and aesthetics of the synthesized images. StyleGAN model is based on a straightforward GAN framework that contains two main networks, namely the generator $\mathcal{G}$ and the discriminator $\mathcal{D}$. Both of them make use of a mapping network $\mathcal{F}$ which helps to disentangle the representation of the noise space.

Figure 3.1 shows an overall view of TC-StyleGAN2. Gray boxes indicate parts that we reuse weights pre-trained from standard StyleGAN2 models. White objects denote deep networks: generator, discriminator, and text encoder. Yellow boxes represent vectors. Orange ones employ (non)linear projections and transformations. Finally, red shapes are scalars. We made two main modifications to the original model in order to introduce natural language information while still being able to reuse the pre-trained weights: (i) The dimensions of the noise vector $\mathbf{z} \in \mathcal{Z}^{512}$ are split in half, and we concatenate text information in the other half of the vector, and (ii) we enforce the discriminator latent representation vector $\mathbf{d}$ to have high cosine similarity to the sentence embedding $\mathbf{s}$ used as the image synthesis condition. Such score is also used as additional information for the discriminator prediction.

Both modifications are important so the generator can synthesize a plethora of different images due to the sampling $\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2)$, though respecting the condition fixed on the other part of the input vector. By adding the cosine constraint in the discriminator, it becomes able to penalize when generated images do not correspond to the original sentence.

3.2 Generator

Formally, our synthesis network takes two input vectors: the noise vector $\mathbf{z} \sim \mathcal{Z}^{256}$ and the textual condition vector $\mathbf{s} \in \mathcal{S}^{256}$. Such vectors are concatenated and then mapped to an intermediate representation $w \in \mathcal{W}^{512}$ through 8 layers of a non-linear mapping network $\mathcal{F}$. Given that we adjusted and projected the input vectors into the regular dimensionality, it is possible to load trained weights for the entire generator $\mathcal{G}$, including $\mathcal{F}$.

Note that z follows a certain probability density and $\mathcal{S}$ is a pre-trained embedding space (see Section 3.5), but the intermediate latent space $\mathcal{W}$ can

learn a more linear, less entangled representation since it does not have a previously defined distribution. The intermediate representation w through learnable affine transformations generates the style codes $\mathbf{t}_\gamma$ and $\mathbf{t}_\beta$ that are responsible for controlling adaptive instance normalization AdaIN. The AdaIN operation does perform channel-wise operations of scale and shift based on the style vectors projected from w and is used in the synthesis network $\mathcal{G}$ at each convolutional layer. The remaining generator architecture closely follows the original implementation.

3.3 Discriminator

In GANs the discriminator $\mathcal{D}$ network is responsible for detecting if an image is real or artificially generated. It is the responsible for generating a gradient signal to the system given that we can assign discrete labels $y \in \{fake, real\}$. Therefore, the discriminator goal is to estimate the probability of a given image being real or not, i.e., $\mathcal{D} = P(y_i|\mathbf{v})$. We modify the discriminator so its prediction also considers the condition vector $\mathbf{s}$.

First, the discriminator can take either a real image or a generated image. Such image is processed by several convolutional layers that output a discriminator latent representation $\mathbf{d}^{512}$. In parallel, we input the condition vector $\mathbf{s}$ into a new randomly trained mapping network that operates in $\mathbb{R}^{256}$. We then employ a linear projection layer $\mathcal{E}$ to generate $\hat{\mathbf{s}}$, which is a 512-dimensional vector. We use such linear mappings so we can get the same representation level and disentanglement from the intermediate space of the generator. We then compute a similarity score $\cos(\hat{\mathbf{s}}, \mathbf{d})$ to encourage $\mathcal{D}$ to approximate the condition distribution. The final output from the discriminator is the weighted sum (λ_C conditional weight, and λ_D unconditional weight) of a neuron and the cosine score, so both values have weight while detecting if an image is not only real but also correspondent to the natural query.

Our two largest modifications are splitting the input vector from the generator and the introduction of a new randomly initialized mapping network that is parallel to the discriminator main network flow. Such a network mainly adds a constraint to the prediction and does not directly affect all the discriminator layers. It does affect them during backpropagation, given that the gradients are estimated from the loss function defined in Equation 3-1, whose prediction was computed from a combination of the cosine constraint and the neuron prediction. We observe that we can get away with both modifications because they do not strongly impact the main flow of the networks, though they provide additional learning signals.

3.4 Loss function

We optimize the discriminator weights θ_D by minimizing the loss function of Equation 3-1 of the predictions for both real and generated images:

$$\Delta\theta_d \frac{1}{m}\sum_{i=1}^{m}\Big[-\mathcal{A}(-\mathcal{D}(\mathbf{v}_i,\mathbf{s}_i))-\mathcal{A}(\mathcal{D}(\mathcal{G}(\mathbf{z}_i,\mathbf{s}_i)))\Big] \tag{3-1}$$

where m is the number of instances in the batch, and $\mathbf{v}_i$ is the i^{th} image drawn from the real data distribution $\mathcal{I}$, and $\mathbf{z}_i$ and $\mathbf{s}_i$ is the noise sampled and the corresponding sentence embedding from $\mathcal{Z}$ for that iteration. Besides, the activation function $\mathcal{A}(x)$ is defined by $softplus(x) = \frac{1}{\beta}\log(1+exp(\beta * x))$, where β is a hyperparameter. Note that that $\mathcal{D}(\mathbf{v}_i,\mathbf{s}_i) = \cos(d,\hat{\mathbf{s}})\lambda_C + P(y_i|\mathbf{v}_i)\lambda_D$, where λ_C and λ_D are the weights for the conditional and unconditional prediction, respectively.

For optimizing the weights θ_g of the synthesis network, we use the opposite of the loss function for the $\mathcal{D}$ as shown in Equation 3-2.

$$\Delta\theta_g \frac{1}{m}\sum_{i=1}^{m}\Big[-\mathcal{A}(\mathcal{D}(\mathcal{G}(\mathbf{z}_i,\mathbf{s}_i)))\Big] \tag{3-2}$$

The overall optimization problem objective is then formulated as the following adversarial training framework:

$$\min_{\mathcal{D}}\max_{\mathcal{G}}\mathbb{E}_{\mathbf{v}\sim\mathcal{I}}[-\mathcal{A}(-\mathcal{D}(\mathbf{v}_i,\mathbf{s}_i))]+\mathbb{E}_{\mathbf{z}\sim\mathcal{Z}}[-\mathcal{A}(\mathcal{D}(\mathcal{G}(\mathbf{z}_i,\mathbf{s}_i)))] \tag{3-3}$$

3.5 Text Encoder

Condition representation: a core aspect of our architecture is the design of the text encoder that will extract a vector representation from the text queries. Such encoder should be able to represent details and fine-grained information from text for the used datasets. Text descriptions were encoded using the Deep Attentional Multimodal Similarity Model (DAMSM) from the AttnGan encoder module (55). The idea of DAMSM module draws inspiration from multimodal alignment models (91, 92), where it learns an image-text encoding function, $\phi(I)$ and $\phi(S)$, that projects both paired representations into the same multimodal space. Such functions are trained so that the distances of related pairs are minimized while unpaired images and texts are far from each other. It does that by training a global representation for images and text while using a cross-attention mechanism to improve on local and fine-rained detail recognition. Note such encoder defines the condition space. Recently, Ye et al.(57) improved AttnGan and DM-GAN (58), using a contrastive learning approach. This technique was employed

in the DAMSM module in the pre-training stage, learning a more consistent textual representation. Then, this method was utilized in the GANs training, enhancing the consistency between the generated images and their respective captions. Experimental results have shown that the quality of synthesized images has improved significantly in terms of FID. In this work, we used the original DAMSM text encoder module for representing the caption embedding.

Text Augmentation: considering the condition space, one can see that if the amount of text queries is limited, we have a discontinued space. In order to increase the continuity and smoothness of such space, we employ the Condition Augmentation (CA) technique (17). Hence, instead of considering the embedding $\phi(S_i) = \mathbf{s}_i$ of each caption for a given image $\mathbf{v}_i$, we sample a textual embedding $\hat{\mathbf{s}} \sim N(\mu(\mathbf{s}), \Sigma(\mathbf{s}))$ where $\mu(\mathbf{s})$ and $\Sigma(\mathbf{s})$ are the mean and diagonal of the covariance matrix of $\mathbf{s}_i$. Such statistics represent the textual embedding distribution for a given image. With the aid of CA, the model is going to take far more training pairs, which is particularly important for small datasets such as CUB and Oxford Flowers.

3.6 Adaptive Discriminator Augmentation

One of the largest challenges in training GANs is the amount of data needed for training some models, such as StyleGAN-size models. In limited datasets, it is easy for the discriminator to overfit the data. Recent work (18) has proposed a mechanism that stabilizes training in limited data regimes using an adaptive discriminator augmentation technique, namely ADA. The technique involves applying 18 types of transformations to the training images with a given probability p. The probability p is adaptively incremented or decremented by a fixed value based on an overfitting level score generated by a heuristic. ADA has proved to be effective in improving the transfer of learning in unconditional GANs, leading to better FID and IS results in several benchmark datasets (93, 13, 94). The default incarnation of TC-StyleGAN2 uses ADA, which proved important in unconditional-to-conditional transfer learning. It allows us to finetune large models in small datasets while maintaining generalization capabilities. We provide a complete ablation regarding its impact in Section 4.1.

3.7 Unconditional-to-Conditional Transfer Learning

TC-StyleGAN2 employs the DAMSM text encoder. To establish a textual metric less prone to overfitting, we incorporated diverse encoders for the

experiment, including the original CLIP version and another version that we fine-tuned specifically for the relevant dataset. Furthermore, Table 4.6 also presents a comparison baseline using real images - instead of conducting the experiment with synthetically generated images, we also used real images.

The CLIP encoder exhibits an R-precision of 10.65% for real images in the CUB-200 Birds dataset and 11.85% in the Oxford-102 Flowers dataset. Its fine-tuned variant yields 26.33% and 22.67% respectively. The TC-StyleGAN2 model shows comparable performance, achieving an R-precision of 25.52% for the birds dataset and 23.45%for the flowers dataset. Consequently, our model generates synthetic images which so closely mirror real images that the retrieval experiment results for both real and artificial images are remarkably similar. This demonstrates that TC-StyleGAN2 generates synthetic images that align exceptionally well with the text, achieving a level of similarity to the real images.

We replicated the same experiment with AttnGAN on CUB-200 Birds dataset, also reporting the R-precision values, summarized in Table 4.7. The R-precision utilizing the DAMSM Encoder was 63.89%, notably higher than the 17.17% for real images, whereas our model demonstrated a score of 23.81% for the task. Frolov et al.(95) and Zhang et al.(69) contend that many generative models tend to overfit the R-precision score, exhibiting R-precision scores that are considerably higher than repeating the same experiment but retrieving captions from real images, thus necessitating further scrutiny. The elevated R-precision score of AttnGAN is comprehensible, given that its error function includes the conditional error $\mathcal{L}_{DAMSM}$, which measures the similarity between text and image using DAMSM encoders. Thus, in the training procedure, AttnGAN also maximizes the alignment between image and text, obtaining a much higher R-precision metric than the experiment for real images.

We also conducted the experiment using the CLIP encoder, which has been trained on 400M text-image pairs. As a result, CLIP comprehends features that extend beyond the specific dataset in question, which can compromise the precision of text-image alignment. However, it was observed that TC-StyleGAN2 yielded R-precision scores of 8.38%, showing results similar to the real images experiment 10.64%. Conversely, AttnGAN demonstrated a notable drop in R-precision of 74.54%, leading to a score of 16.19%. However, this decline was not observed when employing the fine-tuned CLIP encoder, which achieved an R-precision of 60.18%. This suggests that when the image encoder and the text encoder are sufficiently aligned within the specific dataset, AttnGAN exhibits an R-precision that significantly surpasses the score for real images. ne of the main goals of this work is to understand the use of pre-

training information from StylenGAN2, trained in an unconditional paradigm in larger datasets. We expect that by taming such unconditional models into conditional ones, we should be able to generate authentic real-looking images coherent with their respective textual descriptions. This would not only accelerate the training convergence process, but also improve the overall quality of the images. For most of our experiments, we use the StyleGAN2 pre-trained weights on FFHQ (13) data. FFHQ is a far more diverse dataset than CUB and Flowers-102. During training, there must be a domain shift between the source and target images. Notably, we have added extra level of complexity to the model so it can take condition vectors. Hence, we reuse all weights from StyleGAN2-FFHQ to initialize $\mathcal{D}$ and $\mathcal{G}$ weights in the new architecture (see modules inside the gray boxes in Figure 3.1). Some layers had to be randomly initialized, such as the mapping network and linear projection of the condition vector employed in parallel to the discriminator. In Section 4.1 we show that such modifications benefit text-to-image synthesis by using generator and discriminator pre-trained weights from unconditional models.

4
Results

In this chapter, we will describe the experiments we conducted to evaluate the effectiveness of our proposed method. Specifically, we aimed to determine whether our modified architecture, which retains the entire StyleGAN2 architecture with minor modifications for text-conditioned generation, can generate high-quality images. Additionally, we evaluated if reusing pre-trained weights of an unconditional model can stabilize the training convergence, also achieving better results.

In section 4.1, we provide a detailed overview of the experiments we conducted. In section 4.2, we describe the datasets that we used in these experiments. Section 4.3 outlines how we evaluated our proposal method. We present a quantitative analysis of the results in section 4.4, and a qualitative analysis in section 4.5.

4.1
Experiments

To evaluate our text-to-image synthesis model, we used the commonly employed CUB-200 Birds (14) and Oxford-102 Flowers (15) datasets as baselines. Our preliminary experiments focused on identifying the optimal source dataset on which StyleGAN2 was trained to effectively apply transfer learning for generating images within these target baseline datasets. We evaluated four different source datasets: Flickr-Faces-HQ (FFHQ and FFHQU) (13, 41), LSUN-DOGs (96), and CelebA (97). Applying a traditional transfer learning approach between unconditioned models, we investigated whether the pre-existing knowledge captured by the StyleGAN model trained on the source dataset could be transferred to the target dataset, even if the two datasets are from different domains. This enabled us to assess the potential benefits of transfer learning in our text-to-image synthesis model.

Based on this study, we utilized pre-trained weights from the Flickr-Faces-HQ dataset for our U2C transfer approach. We conducted various experiments, which involved exploring the use of text augmentation techniques, implementing the adaptive discriminator augmentation (ADA) technique, and adjusting the hyperparameters λ_D and λ_C.

In Section 4.4, models are evaluated with the standard GAN evaluation metrics and text-alignment metrics. We also showcase qualitative studies in Section 4.5.

4.1.1 Implementation Details

All models were optimized with Adam ($\beta_1 = 0.0$, $\beta_2 = 0.99$) and learning rate of 2×10^{-3} for both discriminator and generator networks. We chose StyleGAN2's R1 regularization weight equal 0.8192, as suggested by guidelines of StyleGAN2 implementation (98).

4.2 Datasets

We evaluated our study on two popular datasets and baselines:

- **Caltech-UCSD Birds (CUB)** (14): The dataset has 200 different categories with 11,788 images of birds in total. Each image contains 10 text descriptions of bird characteristics. CUB is split in 8,855 images of 150 categories for training and 2,933 images of 50 categories for testing.
- **Oxford-102** (15): The dataset has 102 different categories with 8,189 images of flowers in total. Each image contains 10 text descriptions of flowers characteristics. Oxford-102 is split in 7,034 images for training and 1,154 images for testing.

We conducted experiments utilizing pre-trained StyleGAN2 weights from a variety of datasets, but more diverse and complex than the baseline ones. We considered the Flickr-Faces-HQ (FFHQ) (13) dataset, the Flickr-Faces-HQ-U (FFHQ-U) (41), the LSUN-DOGs (96), and the CelebA (97).

4.3 Evaluation

To evaluate the effectiveness of our study, we employed three commonly used metrics in generative image models: the Inception Score (IS) (99), the Fréchet Inception Distance (FID) (43), and the Kernel Inception Distance (KID) (68), as they provide quantitative measures of image quality. For the FID and KID metrics, we followed the approach used in the StackGAN++ study and evaluated them based on all the test captions and statistics from the training data.

In addition, we evaluated the alignment quality between the text and visual data using several metrics: the R-precision score, the Mean Reciprocal Rank (MRR), Precision, and our newly proposed metric, which allowed us to measure how well the generated images matched their corresponding captions.

4.4 Quantitative Analysis

In this section, we present the outcomes of our experiments. In Sub-Section 4.4.1, we elaborate on a previous analysis conducted to evaluate the feasibility of transfer learning between non-conditional models with entirely dissimilar domains in terms of target and source datasets. Sub-section 4.4.2 compares our TC-StyleGAN2 method to state-of-the-art GANs, showcasing improvements in stability and convergence, as well as achieving superior results in fewer iterations. Lastly, in 4.4.3, we investigate the textual alignment between the generated images and their corresponding input queries.

4.4.1 Unconditional Experiments

First, we aim to determine if transfer learning between distinct domains enhances the training of non-text-conditioned GANs. Specifically, we investigate if leveraging prior knowledge from a more diverse, less constrained, and complex dataset, trained for an extended period, provides any benefits. In the least favorable scenarios, if the initial network configuration is not advantageous, the network would need to disregard all irrelevant prior knowledge, potentially resulting in slower learning. Alternatively, the incorrect initialization could lead to convergence issues, ultimately generating unreliable images.

In our initial experiment, we utilized pre-training data from StyleGAN2 that had been trained on the FFHQ human faces dataset. We then proceeded to train the same network to generate images of birds and flowers. The training progress can be observed in Figure 4.1, which illustrates the FID score over time, measured in terms of iterations. The left figure depicts the training progression for the Oxford-102 Flowers dataset, while the right one represents the CUB-200 Birds dataset. Notably, we observed a remarkable drop in the FID score during the early iterations, indicating a significant improvement in the generated images' quality in the early stages. This trend was consistent for both the flowers and birds datasets. Subsequently, the training process stabilized in the initial iterations, suggesting a convergence towards a more optimal state.

We also conducted a comparative analysis against a baseline approach (training from scratch) using the FFHQ, CelebA, and LsunDogs datasets. The findings are summarized in Figure 4.2, which presents the FID score over time, measured in iterations. We observed that leveraging pre-training information, even from vastly different datasets, proved advantageous for stabilizing the training process in non-conditional transfer learning. Building

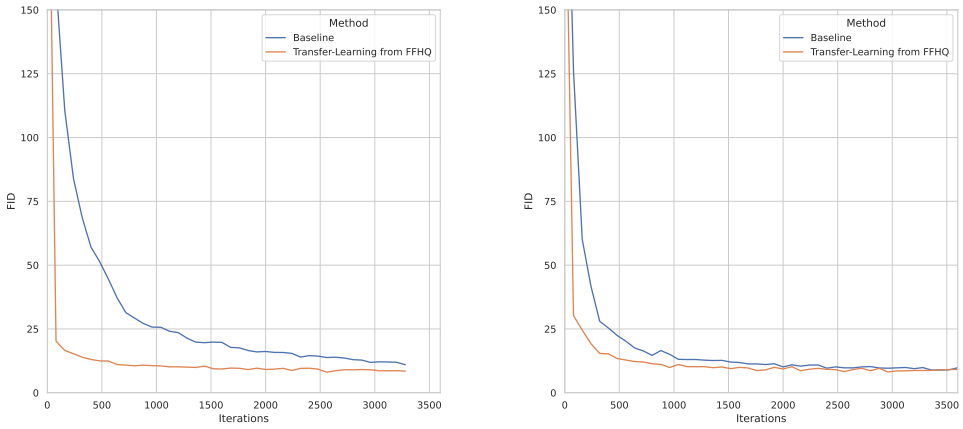

Figure 4.1: Training progress comparative of the original and unconditioned StyleGAN2, in terms of FID (lower is better). Comparison between training from scratch (baseline) and using pre-trained weights from FFHQ. The leftmost figure shows the Oxford-102 Flowers results, and the rightmost shows the CUB-200 Birds results.

upon this observation, our investigation aimed to determine whether this benefit extends from non-conditional to text-conditioned models, even when applied to architecturally different networks but designed to be as similar as possible. For this, we chose to proceed with the FFHQ faces dataset, as it is the most divergent of the target baseline datasets, demonstrating a greater contrast of domains.

4.4.2 Baselines

To evaluate the efficiency of our training and transfer-learning strategies, we trained our models for a maximum of 24 hours using a single v100 GPU per run. We used $\{\lambda_D = 0.25, \lambda_D = 1\}$ for CUB-200, and $\{\lambda_D = 1, \lambda_D = 1\}$ for Oxford-102. The pre-training information used was based on the StyleGAN2 FFHQ architecture.

We compare TC-StyleGAN2 to state-of-the-art approaches, such as RATGAN (100), Lightweight ManiGAN (101), Souza, Wehrmann e Ruiz(4) and LAFITE (64), and a baseline that is the same architecture of TC-StyleGAN2 but without Unconditional-to-Conditional Transfer Learning.

Table 4.1 shows quantitative results for our primary approach, TC-StyleGAN2, as well as results from prior work. Notably, our approach outperformed all past work by large margins. We achieve 7.48 FID versus 8.02 for Lightweight ManiGAN in CUB dataset. The third best result is 10.48 FID from LAFITE. Note that TC-StyleGAN2 results present a relative improve-

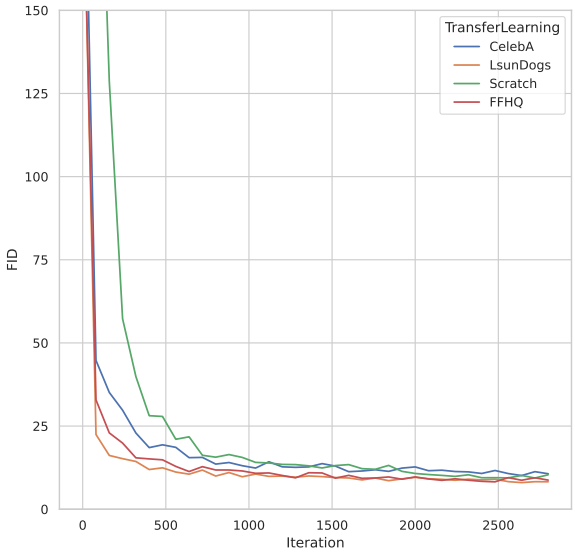

Figure 4.2: Training progress comparative of the original and unconditioned StyleGAN2, in terms of FID. The evaluation includes a comparison between training from scratch (baseline) and utilizing pre-trained weights from FFHQ, CelebA, and LsunDogs datasets, with the experiments conducted on the CUB-200 dataset.

ment of roughly 40% when compared to LAFITE. Compared to the previously reported FID results for Oxford-102, once again results of TC-StyleGAN2 are second to none. For the sake of completeness, we also report Inception Scores (IS), which do help to confirm that our approach improves over the prior art for CUB dataset. We highlight that that IS values are not as reliable as FID ones. Results clearly show that IS are not as efficient in measuring progress in the field as FID ones. For instance, when we compare StackGAN++ (older model that produces lower quality samples) to our approach the relative improvement of FID for Oxford-102 is 650%; while IS values show only a 17% improvement, and most of the remaining models actually fall in the standard variation range.

Recall that standard TC-StyleGAN2 incarnation is built on top of the StyleGAN2 architecture and employs unconditional-to-conditional transfer learning, Condition Augmentation, and ADA. Table 4.2 provides an ablation study that shows the impact of each component in the generator design of TC-StyleGAN2. The models trained from scratch (all layers randomly initialized) had the worst results, and sometimes the training procedure diverged. ADA clearly causes substantial reductions of **33.05%** and **66.99%** in the FID scores. It is also very clear that Unconditional-to-Conditional Transfer Learning does bring important improvements to the results. Textual augmentation improved

	FID ↓		IS ↑	
Methods	CUB	Oxford-102	CUB	Oxford 102
StackGAN++ (2)	15.30	48.68	4.04 ± 0.05	3.26 ± 0.01
AttnGAN (55)	23.98	-	4.36 ± 0.03	-
DM-GAN (58)	16.09	-	4.75 ± .07	-
RATGAN (100)	13.91	-	5.36 ± 0.20	**4.09**
Souza, Wehrmann e Ruiz(4)	11.17	16.47	4.23 ± 0.05	3.71 ± 0.06
Lightweight ManiGAN (101)	8.02	-	-	-
LAFITE (64)	10.48	-	5.97	
TC-StyleGAN2 (Ours)	**7.49**	**9.47**	**5.99 ± 0.20**	3.84 ± 0.15

Table 4.1: Comparison of TC-StyleGAN2 against state-of-the-art models.

	CUB-200		Oxford-102	
Methods	FID ↓	KID (x 10^3) ↓	FID ↓	KID (x 10^3) ↓
From Scratch	14,04	5,55	34,44	22,87
+ ADA	9,40	4,11	11,37	3,51
+ U2C transfer	8,02	2,25	**9,47**	**2,11**
+ Conditional Augmentation	7,53	**2,07**	10,13	2,87
+ λ_D Tuning	**7,49**	2,14	9,85	2,41

Table 4.2: FID and KID for various generator designs (lower is better)

performance on the CUB dataset, while for Oxford-102 Flowers dataset results decreased marginally.

In Figure 4.3, training from scratch results in more training instability after a few iterations, and may cause divergence. Hence, each proposed component in TC-StyleGAN2 is quite important and helpful not only for improving generalization but also for accelerate and stabilize the training procedure. Using ADA the FID will take longer to improve but model gets far more robust to overfitting and unstability. Using the complete approach (ADA+Finetuning), the transfer-learning results in state-of-the-art FID values very quickly. As demonstrated in Figure 4.4, our approach surpasses most existing methods within a few hundred iterations. Furthermore, even when training from scratch, without leveraging transfer learning or any of the augmentation techniques we have discussed, our standalone method exhibits promising results.

Table 4.3 provides a quantitative analysis of the outcomes achieved through various generator models. Each model was trained using a range of configurations, which included initializing all layers from scratch, implementing the ADA mechanism, or applying the U2C transfer method. It is important to note that text augmentation was not utilized in these experiments. We standardized the design by setting both λ_C and λ_D to 1 for all models. Our findings suggest that the ADA mechanism is beneficial in promoting both training progression and quality metrics. Additionally, we found that by incorporating U2C transfer as an initialization procedure, we could

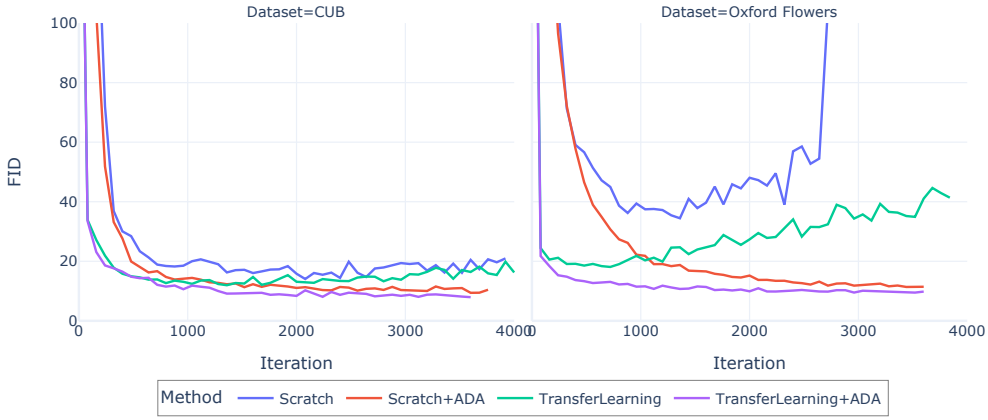

Figure 4.3: FID for various generator designs (lower is better)

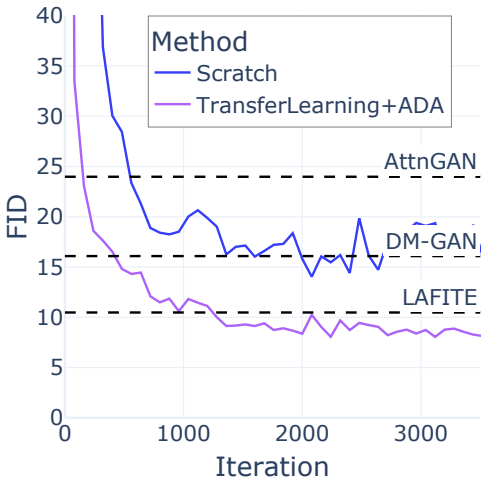

Figure 4.4: TC-StyleGAN2 outperforms most of the prior work in a few hundred iterations.

significantly enhance the FID scores. This approach led to a 175% improvement for the CUB 200 dataset and a 363% improvement for the Oxford-102 Flowers dataset. The data on KID scores, displayed in Table 4.4, mirror these findings. There is a considerable decrease in the KID scores when we implement our full method (TC-StyleGAN2 +ADA), indicating a substantial improvement in image quality.

Table 4.5 presents quantitative results for the default incarnation of TC-StyleGAN2, which employs the ADA mechanism. Additionally, we integrated a text-augmentation mechanism with the conditional parameter λ_D set to fluctuate between 0 and 1, while maintaining $\lambda_C = 1$ as a constant. Our study concluded that optimal results for the CUB dataset were obtained with $\lambda_C = 0.25$, and for the Oxford-102 dataset, $\lambda_C = 0.5$ yielded the best results. The λ_C parameter carries significant weight in the error function that measures the alignment with the text. As a result, fine-tuning this parameter is crucial

	From Scratch		Ours (U2C transfer)	
Dataset	+ADA	-ADA	+ADA	-ADA
CUB	9,40	14,04	**8,02**	11,96
Oxford-102	11,37	34,44	**9,47**	18,09

Table 4.3: FID scores considering different training configurations (lower is better)

	From Scratch		Ours (U2C transfer)	
Dataset	+ADA	-ADA	+ADA	-ADA
CUB	4,11	5,55	**2,25**	5,60
Oxford-102	3,51	22,87	**2,11**	9,25

Table 4.4: KID scores considering different training configurations (lower is better)

to enable the model to generate realistic images that adhere closely to the corresponding text. When λ_C is set to 0, the training process does not benefit from reinforcement of learning the conditional part, which consequently leads to weaker textual alignment with the produced image.

Dataset	λ_D	λ_C	FID ↓	KID (x 10^3) ↓
CUB	0,125	1	7,61	2,20
	0,25	1	**7,49**	2,14
	0.75	1	7,83	**1,97**
	1	1	7,53	2,07
Oxford-102	0,125	1	10,00	2,43
	0,25	1	9,95	**2,39**
	0.5	1	**9,85**	2,41
	0.75	1	10,36	2,64
	1	1	10,13	2,87

Table 4.5: FID and KID scores (lower is better) for λ tunning

4.4.3 Text Alignment analysis

In this Sub-Section, we will evaluate the images produced by text-conditioned generative models not merely based on their realism but also their congruity with the given input query. Our analysis will specifically focus on this aspect of textual alignment.

Table 4.6 provides a comparative analysis of the retrieval experiment conducted. The experimental procedure is as follows: the model generated an image for each caption in the test set, and then 99 non-corresponding captions were randomly selected. Subsequently, we attempted to retrieve the correct caption from a pool of 100 captions (99 incorrect and one correct) for the generated image, utilizing both the image and textual encoder, with cosine

	Fake Images		Real Images	
Dataset	Clip	Clip fine-tuned	Clip	Clip fine-tuned
CUB-200 Birds	8,38	25,52	10,65	26,33
Oxford-102 Flowers	9,3	23,45	11,85	22,67

Table 4.6: R-precision (%) for real and fake images, using original CLIP and its version fine-tuned on the correspondent dataset.

similarity serving as our distance metric. Essentially, the table encapsulates the accuracy rate of correct retrievals, indicating instances where the caption prompted from the image generation model indeed matched the caption retrieved in each experimental iteration.

TC-StyleGAN2 employs the DAMSM text encoder. To establish a textual metric less prone to overfitting, we incorporated diverse encoders for the experiment, including the original CLIP version and another version that we fine-tuned specifically for the relevant dataset. Furthermore, Table 4.6 also presents a comparison baseline using authentic images - instead of experimenting with synthetically generated images, we also used authentic images.

The CLIP encoder exhibits an R-precision of 10.65% for real images in the CUB-200 Birds dataset and 11.85% in the Oxford-102 Flowers dataset. Its fine-tuned variant yields 26.33% and 22.67%, respectively. The TC-StyleGAN2 model shows comparable performance, achieving an R-precision of 25.52% for the Bird's dataset and 23.45% for the Flowers dataset. Consequently, our model generates synthetic images that so closely mirror real images that the retrieval experiment results for authentic and artificial images are remarkably similar. Therefore, our TC-StyleGAN2 generates synthetic images that align exceptionally well with the text, achieving a level of similarity to the real images.

We replicated the same experiment with AttnGAN on CUB-200 Birds dataset, also reporting the R-precision values, summarized in Table 4.7. The R-precision utilizing the DAMSM Encoder was 63.89%, notably higher than the 17.17% for authentic images, whereas our model demonstrated a score of 23.81% for the task. Frolov et al.(95) and Zhang et al.(69) contend that many generative models tend to overfit the R-precision score, exhibiting R-precision scores that are considerably higher than repeating the same experiment but retrieving captions from real images, thus necessitating further scrutiny. The elevated R-precision score of AttnGAN is comprehensible, given that its error function includes the conditional error $\mathcal{L}_{DAMSM}$, which measures the similarity between text and image using DAMSM encoders. Thus, in the training procedure, AttnGAN also maximizes the alignment between image and text, obtaining a much higher R-precision metric than the experiment for

Method	DAMSM Encoder	Clip Encoder	Clip fine-tuned
AttnGAN	63,89	16,19	60,18
TC-StyleGAN2	23,81	8,38	25,56
Real Images	17,17	10,64	26,39

Table 4.7: R-precision(%) for real and fake images, using original CLIP encoder and its version fine-tuned on the correspondent dataset, comparing our method with AttnGAN.

Dataset	Images	MRR	R-Precision (%)	Precision@3 (%)	Precision@5 (%)
CUB	Ours	0.425	25.56	50.32	63.53
	AttnGAN	0.74	60.18	85.81	93.08
	Real Images	0.427	26.39	49.96	62.17
Oxford-102 Flowers	Ours	0.397	22.96	46.17	59.20
	Real Images	0.40	22.63	47.82	62.74

Table 4.8: R-precision, Mean Reciprocal Rank, Precision@3 and Precision@5 for TC-StyleGAN2, AttnGAN and Real Images on CUB-200 Birds and Oxford-102 Flowers dataset, using Clip fine-tuned as encoder.

authentic pictures.

We also experimented using the CLIP encoder, trained on 400M text-image pairs. As a result, CLIP comprehends features that extend beyond the specific dataset in question, which can compromise the precision of text-image alignment. However, TC-StyleGAN2 yielded R-precision scores of 8.38%, showing results similar to the authentic images experiment 10.64%. Conversely, AttnGAN demonstrated a notable drop in R-precision of 74.54%, leading to a score of 16.19%. We did not observe the decline when employing the fine-tuned CLIP encoder, which achieved an R-precision of 60.18%, suggesting that when the image encoder and the text encoder are sufficiently aligned within the specific dataset, AttnGAN exhibits an R-precision that significantly surpasses authentic image scores.

Table 4.8 encapsulates the experiments conducted, showcasing the R-precision, the Mean Reciprocal Rank, the Precision@3 and the Precision@5 for our method TC-StyleGAN2 and the AttnGAN, and employing the fine-tuned CLIP as both the textual and image encoder. The outcomes of our method for both datasets align closely with the results derived from real images, underscoring that the generated synthetic images bear a striking resemblance to the distribution of the actual images.

A compatibility metric between the synthesized image and the corresponding text is used as a conditional error during the training procedure in many generative image models. Therefore, aligning metrics that quantify the disparity between image and text often produces high results. We propose a novel metric explicitly designed to evaluate the alignment between textual de-

Method	Dataset	Clip	Clip fine-tuned
Ours	CUB-200 Birds	0.8840	0.9422
	Oxford-102 Flowers	0.8969	0.9490
AttnGAN	CUB-200 Birds	0.8941	0.9483

Table 4.9: Cosine similarity metric adjusted between a real image and a synthetic image, using Clip and Clip fine-tuned in the respective dataset as image encoders.

scriptions and images, utilizing authentic images for this purpose. For every artificially generated image produced by the model, we compute its adjusted cosine similarity with the corresponding real image. This computation enables us to quantify the extent of deviation between the real and artificially produced images, thus indirectly assessing the alignment between the generated image and its textual description. We express the reported metric as the average and standard deviation of all distances from the artificially generated images to their corresponding real counterparts.

Equation 4-1 defines the metric $M(\mathcal{G}, \mathbf{x}_{data})$, which is calculated by averaging the adjusted cosine similarities between the encodings of the real and generated images across the entire test dataset. In this equation, N denotes the total number of captions in the test dataset, which is denoted by $\mathbf{x}_{data}$. The function ϕ symbolizes the image encoder that is applied to each image. Each s_i represents the $i-th$ caption or sentence in the test dataset, and each z_i, is the random noise drawn from the $\mathcal{Z}$ distribution that is used for image generation. The real image that corresponds to the $i-th$ sentence is denoted as $\boldsymbol{real}_i$.

$$M(\mathcal{G}, \mathbf{x}_{data}) = \frac{1}{2N} \sum_{i=1}^{N} \frac{\phi(\boldsymbol{real}_i).\phi(G(\mathbf{z}_i, \mathbf{s}_i)}{||\phi(\boldsymbol{real}_i)||\phi(G(\mathbf{z}_i, \mathbf{s}_i)||} + 1 \tag{4-1}$$

The metric discussed in Table 4-1, calculated via the original CLIP encoder and a CLIP encoder fine-tuned for the specific dataset, provides a summary of the synthetic images created using our suggested method and the AttnGAN architecture. It's important to note that the metric scores from our method are pretty similar to those obtained from AttnGAN, regardless of the encoder employed. For the CUB-200 Birds dataset, our method TC-StyleGAN2delivers scores of 0.8840 and 0.9422 when using the CLIP and fine-tuned CLIP encoders, respectively. Likewise, for the Oxford-102 Flowers dataset, our method achieves scores of 0.8969 and 0.9490 with these encoders. In future research, we aim to probe further into the scope of this metric. We intend to understand the impact of a 1% difference and the effect of using different encoders that might not necessarily correlate directly with the text.

Turning our focus to the learned models of text and image representa-

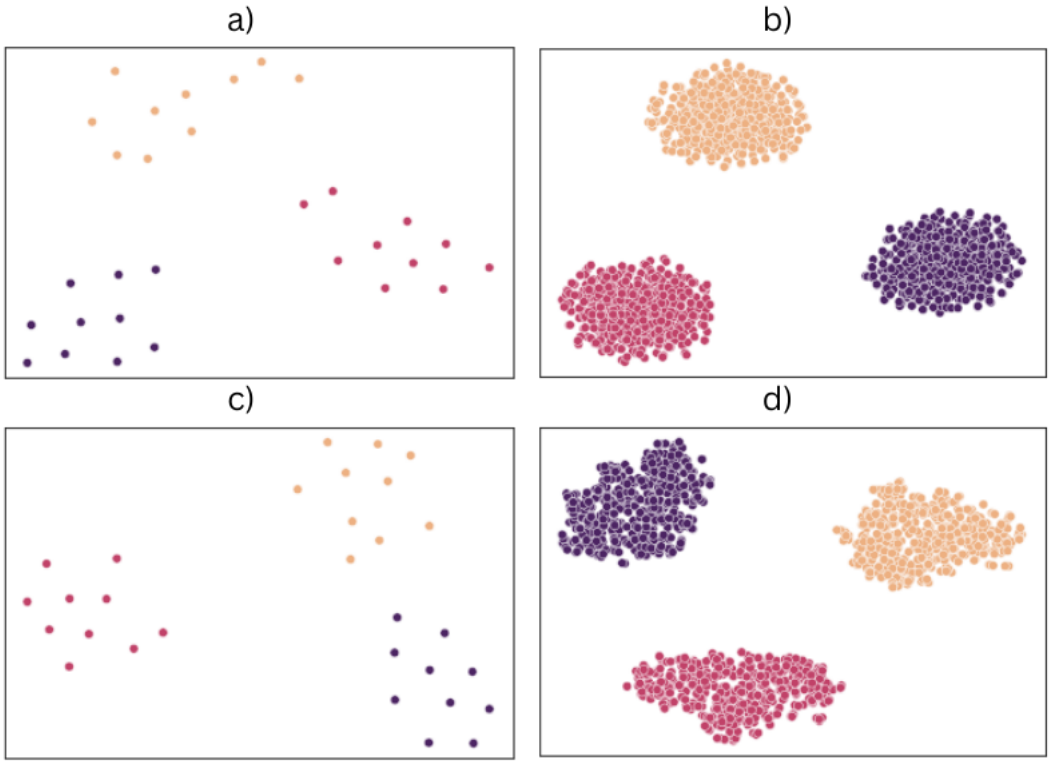

Figure 4.5: Visualization of sentences embbeddings. We sampled 3 random images, and applyed t-SNE (1) to reduce the original space to $\mathcal{R}^2$. In **a)** is shown 30 sentences embeddings, as each image has 10 captions, projecting $\mathcal{R}^{256}$ to $\mathcal{R}^2$. In **b)** is shown 500 sentences embeddings for each image, using the conditional-augmentation module. In **c)** is shown the intermediate representation in $\mathcal{W}$ space, using the 30 captions, projecting $\mathcal{R}^{512}$ to $\mathcal{R}^2$. In **d)** is shown the intermediate representation in $\mathcal{W}$ space of 500 sentences for each image, using the conditional-augmentation module, projecting $\mathcal{R}^{512}$ to $\mathcal{R}^2$.

tion, it is important to note the impact of StyleGAN's Mapping layer and the Conditional Augmentation on the learning process. Figure 4.5 shows a visualization of sentence representation, applying t-SNE (1) to reduce the original space to $\mathcal{R}^2$, accentuating the role of the conditional augmentation module and the Mapping Network layers. As demonstrated in panels b) and d), the application of conditional augmentation allows the generation of a larger set of sentence embeddings than the number of images, thus enabling continuous sampling of the textual representation. Panels c) and d) exhibit a learned representation of the sentence embedding within the $\mathcal{W}$ space. With each image group distinctly separated, the $\mathcal{W}$ space embodies a meaningful and learned representation of the sentence.

Consequently, it is evident that the process of conditional augmentation not only facilitated the generation of additional text-image pairs, mitigating the discontinuity of the highly sparse sentence space, but the Mapping layer also effectively projected this representation into the equally significant inter-

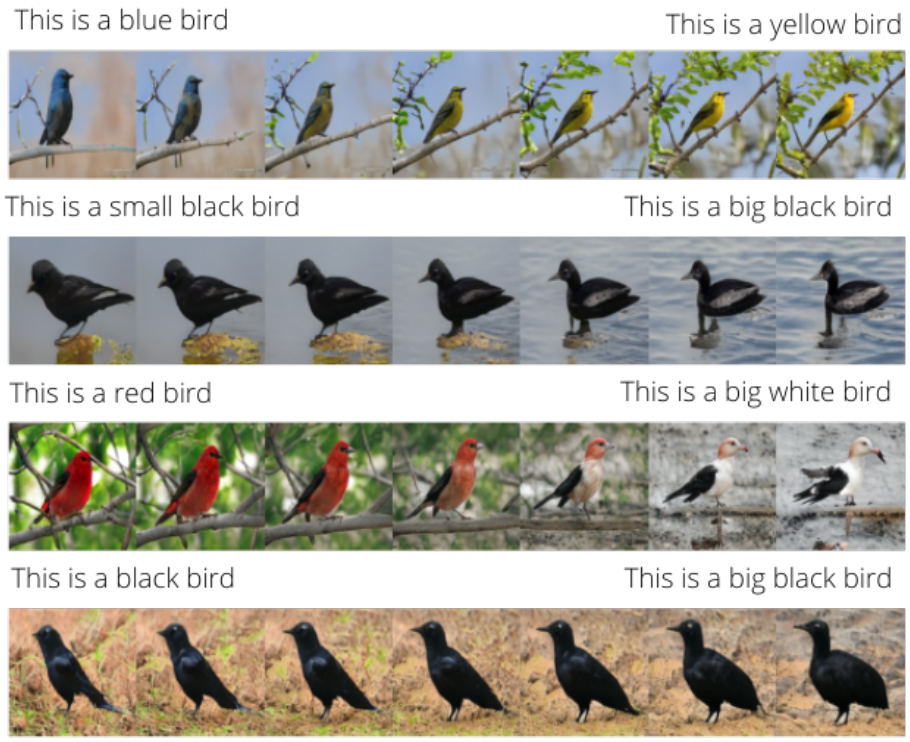

Figure 4.6: Images generated by linear interpolation in the intermediate space $\mathcal{W}$ (left to right).

mediate space. In section 4.5, we illustrate that our model enables interpolation both in the sentence space and the intermediate space, generating intermediary images with reliable features. It also allows the execution of arithmetic operations using these embeddings to either add or subtract attributes. This showcases the effectiveness of the path length regularization technique and other mechanisms employed in the training process of StyleGAN2 in creating a less entangled space $\mathcal{W}$.

4.5 Qualitative Analysis

In this section, we will present qualitative experiments illustrating the efficacy of our method. We will demonstrate the impact of the techniques employed to make the intermediate space $\mathcal{W}$ less entangled. This will be evidenced by the ability to interpolate different embeddings, representing varying sentences, thereby enabling the blending of gradual characteristics. We will also show that interpolation can be performed in both the $\mathcal{Z}$ space and the $\mathcal{W}$ space. Moreover, we will illustrate the feasibility of image manipulation via embedding arithmetic.

Image manipulation: similarly to Souza, Wehrmann e Ruiz(4), our model is also capable of editing images in textual space while preserv-

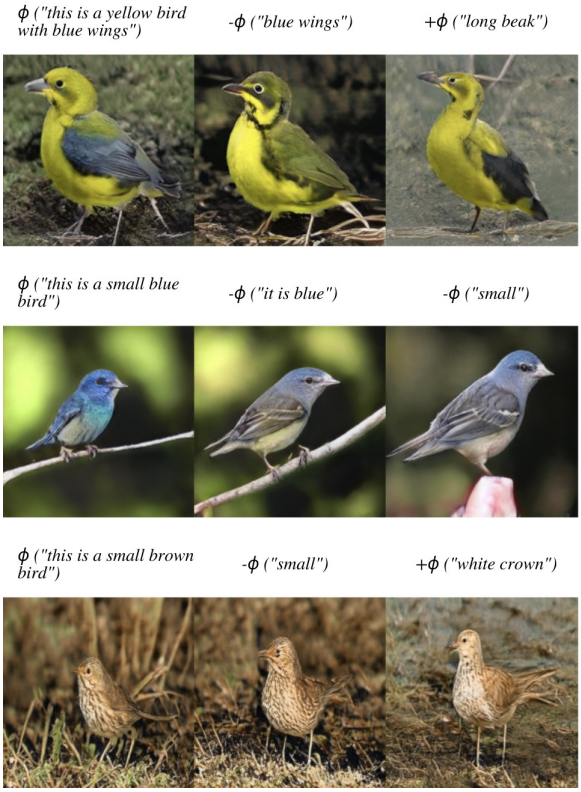

Figure 4.7: Arithmetic in the text-embedding space which enables natural language-based image-editing.

ing structural and main features of the birds and also the environment. The middle illustration in Figure 4.6 shows different examples, with addition and subtraction of characteristics that vary in color, size, and fine-grained details such as beak size and wing color. The sentence encoded by $\phi(\textit{"this is a yellow bird with blue wings"})$ when subtracted from the embedding ϕ("blue wings"), produces an image that preserves its structure, but with the semantic characteristic removed.

Interpolation: one can visualize if the learned $\mathcal{W}$ has structural regularity by interpolating between two vectors in that space. Figure 4.7 shows the interpolation between distinct input condition embeddings but with the same noise. We can observe a gradual merging of features between the generated images, as requested by the prompted text query. It clearly shows that our models were, in fact, learning to respect the condition during Unconditional-to-Conditional Transfer Learning. Even environment details were added smoothly in a semantically meaningful fashion. The water background is gradually added to the image in the second row to match the bird species' living environment. This elucidates how the intermediate latent space has fewer entangled regions than the latent spaces of noise and embedding.

Though, we leave for future work to further explore how to isolate better background modifications when those are not present in the natural queries.

Color Interpolation: the leftmost illustration of Figure 4.8 depicts the interpolation in the latent space of the encodings in 4 different directions: ϕ(*"this is a brown bird"*) (upper left), ϕ(*"this is a red bird"*) (upper right"), ϕ(*"this is a black bird"*) (bottom left) and ϕ(*"this is a yellow bird*) (bottom right). The intermediate images have mixed characteristics of the interpolation direction while, in general, the bird's environment and color characteristics are preserved. Similarly, the rightmost illustration of Figure 4.8 shows an image interpolation in Oxford-102 Flowers data, where the corners are images representing *"A pink flower"* (upper left), *"A yellow flower"(upper right)*, *"An orange flower"* (bottom left) and *"A violet flower"* (bottom right).

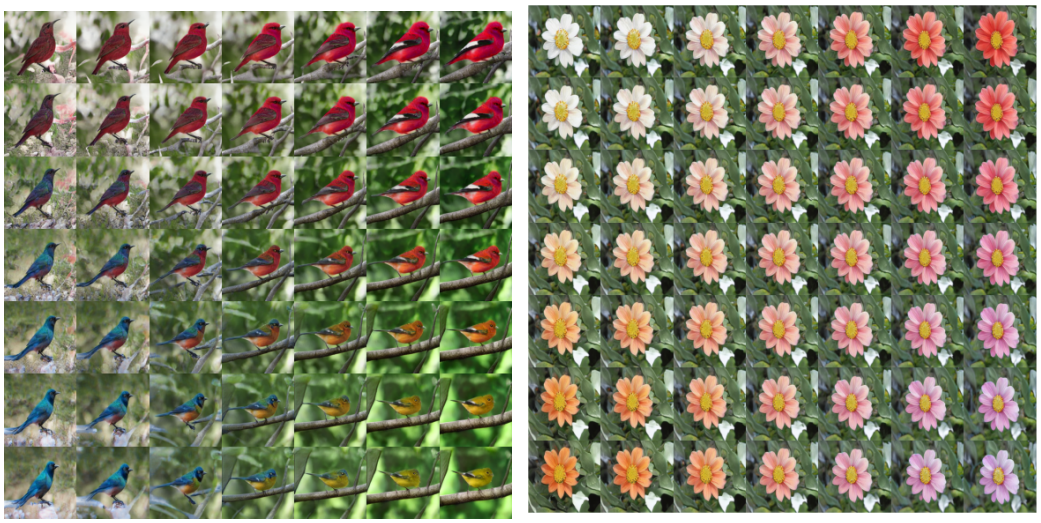

Figure 4.8: Image interpolation in four directions of the text-embedding space of DAMSM text-encoder

Figures 4.9 and 4.10 depict images produced by our method, with their respective textual descriptions, compared to other architectures. Our generated images are highly photo-realistic, with fine-grained details, presenting a coherent semantic correspondence with the captions. Our method produces more realistic backgrounds between the architectures and is more coherent concerning the input query. For example, our method generates all colors in different parts of the petals considering the whole query *"The petals of the flowers are in various colors such as red, green, and purple"* but the other models synthesized images considering only part of this input.

Figure 4.9: Images synthesized by StackGan++ (2), HDGAN (3), Souza, Wehrmann e Ruiz(4) and our method.

4.5.1 Training Progress

Figures 4.11 and 4.12 depict images generated by varying designs at the onset of training. These images were generated at the model's checkpoint after every 80k images viewed, starting from 0.

With the application of the U2C transfer method, the model can generate images of birds and flowers right at the beginning of the training process. In contrast, other model approaches are still producing blurry outputs at this stage. Nonetheless, even within these blurred images, it is discernible that the formation of the images is text-conditioned.

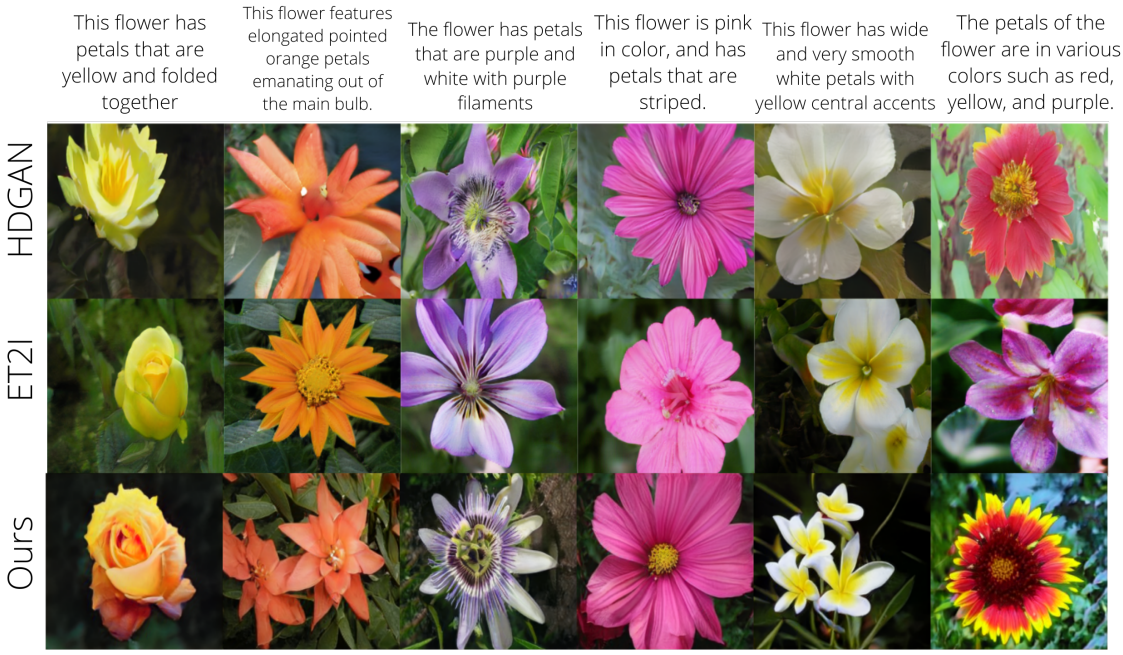

Figure 4.10: Images synthesized by HDGAN (3), Souza, Wehrmann e Ruiz(4) (4) and our method.

Figure 4.11: Fake Images generated by the query "This flower is pink and white in color, with petals that are connected" every 80k images seen in the training, with different generator designs.

Figure 4.12: Fake Images generated by the query "This is a red bird." every 80k images seen in the training, with different generator designs.

5
Conclusion and Future Work

In this work, we proposed a simple yet very effective transfer-learning approach for training text-conditioned GANs, namely Unconditional-to-Conditional Transfer Learning (U2C transfer). By using such an approach, we were able to modify the unconditional architecture of StyleGAN2 to allow text-conditioned image synthesis, which we called Text-Conditioned StyleGAN2 (TC-StyleGAN2).

We also added stronger augmentation recipes and strategies, which allowed us to train reasonably large models in very small datasets. Such a method effortlessly outperformed previous state-of-the-art models by large margins in terms of FID in widely used benchmarks. We have shown that pre-training information of an unconditional model trained in a different and more diverse dataset is beneficial when training in smaller datasets. TC-StyleGAN2 took only a few hundred iterations to top most of the prior work.

In addition, the learning procedure was much more stable when used the proposed strategy. We show that our model is capable of image editing by doing arithmetic operations on the text embedding information and interpolation in the latent intermediate space of the Mapping Network.

Our findings illustrate that the images synthesized using our proposed technique are credible and exhibit strong alignment with their corresponding textual descriptions. In fact, the textual alignment scores achieved by our method are impressively comparable to those of authentic images.

In our future research, we aim to delve into an adaptive version of U2C transfer that can dynamically adjust the currently fixed hyperparameters during training. This adjustment would be based on heuristics regarding the degree of text-conditioning in the model.

Additionally, we plan to broaden our evaluations to include larger and more diverse datasets, which are generally more challenging for methods like StyleGAN2. These models often find it difficult to generate believable scenes under such conditions.

Furthermore, we aim to investigate the application of U2C transfer to other models, to enhance the general applicability of this technique. The architecture implementation we proposed for TC-StyleGAN2 is advantageous in enabling us to convert unconditioned models to conditioned ones easily.

Also, we aim to delve deeper into the alignment between the generated images and the input queries. As numerous models are now producing increasingly lifelike images, gaining a comprehensive understanding of this alignment

is crucial. This will allow us to evaluate these models from a perspective that is free from overfitting and bias. Through these endeavors, we aim to push the boundaries of artificial image generation, fostering advancements in research and practical applications.

6
Bibliography

1 MAATEN, L. van der; HINTON, G. Viualizing data using t-sne. **Journal of Machine Learning Research**, v. 9, p. 2579–2605, 11 2008. Cited 2 times in pages 9 and 50.

2 ZHANG, H. et al. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. **CoRR**, abs/1710.10916, 2017. Disponível em: <http://arxiv.org/abs/1710.10916>. Cited 5 times in pages 9, 23, 24, 44, and 54.

3 ZHANG, Z.; XIE, Y.; YANG, L. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In: **CVPR**. [S.l.: s.n.], 2018. Cited 5 times in pages 9, 25, 29, 54, and 55.

4 SOUZA, D. M.; WEHRMANN, J.; RUIZ, D. D. Efficient neural architecture for text-to-image synthesis. **CoRR**, abs/2004.11437, 2020. Disponível em: <https://arxiv.org/abs/2004.11437>. Cited 8 times in pages 9, 23, 25, 42, 44, 51, 54, and 55.

5 GOODFELLOW, I. J. et al. **Generative Adversarial Networks**. arXiv, 2014. Disponível em: <https://arxiv.org/abs/1406.2661>. Cited 5 times in pages 13, 16, 17, 19, and 20.

6 REED, S. et al. **Generative Adversarial Text to Image Synthesis**. 2016. Cited 3 times in pages 13, 23, and 24.

7 KARRAS, T. et al. **Analyzing and Improving the Image Quality of StyleGAN**. arXiv, 2019. Disponível em: <https://arxiv.org/abs/1912.04958>. Cited 2 times in pages 13 and 14.

8 BROCK, A.; DONAHUE, J.; SIMONYAN, K. **Large Scale GAN Training for High Fidelity Natural Image Synthesis**. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1809.11096>. Cited 2 times in pages 13 and 20.

9 RAMESH, A. et al. **Hierarchical Text-Conditional Image Generation with CLIP Latents**. 2022. Cited 2 times in pages 13 and 21.

10 SAHARIA, C. et al. **Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding**. 2022. Cited 2 times in pages 13 and 21.

11 YU, J. et al. **Scaling Autoregressive Models for Content-Rich Text-to-Image Generation**. 2022. Cited 2 times in pages 13 and 21.

12 WANG, K. et al. Pay attention to features, transfer learn faster cnns. In: **International Conference on Learning Representations**. [s.n.], 2020. Disponível em: <https://openreview.net/forum?id=ryxyCeHtPB>. Cited in page 13.

13 GITHUB - NVlabs/ffhq-dataset: Flickr-Faces-HQ Dataset (FFHQ). <https://github.com/NVlabs/ffhq-dataset>. (Accessed on 07/02/2022). Cited 7 times in pages 13, 21, 22, 36, 38, 39, and 40.

14 WAH, C. et al. The caltech-ucsd birds-200-2011 dataset. In: . [S.l.: s.n.], 2011. Cited 5 times in pages 13, 21, 22, 39, and 40.

15 NILSBACK, M.-E.; ZISSERMAN, A. Automated flower classification over a large number of classes. In: **2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing**. [S.l.: s.n.], 2008. p. 722–729. Cited 5 times in pages 13, 21, 22, 39, and 40.

16 ARJOVSKY, M.; BOTTOU, L. **Towards Principled Methods for Training Generative Adversarial Networks**. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1701.04862>. Cited in page 14.

17 ZHANG, H. et al. **StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1612.03242>. Cited 4 times in pages 14, 23, 24, and 36.

18 KARRAS, T. et al. **Training Generative Adversarial Networks with Limited Data**. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2006.06676>. Cited 4 times in pages 14, 30, 31, and 36.

19 WIATRAK, M.; ALBRECHT, S. V.; NYSTROM, A. **Stabilizing Generative Adversarial Networks: A Survey**. arXiv, 2019. Disponível em: <https://arxiv.org/abs/1910.00927>. Cited 3 times in pages 17, 18, and 20.

20 NOWOZIN, S.; CSEKE, B.; TOMIOKA, R. **f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1606.00709>. Cited in page 17.

21 MAO, X. et al. **Least Squares Generative Adversarial Networks**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1611.04076>. Cited in page 17.

22 ARJOVSKY, M.; CHINTALA, S.; BOTTOU, L. **Wasserstein GAN**. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1701.07875>. Cited 2 times in pages 17 and 18.

23 ZHOU, Z. et al. **Lipschitz Generative Adversarial Nets**. arXiv, 2019. Disponível em: <https://arxiv.org/abs/1902.05687>. Cited in page 18.

24 GULRAJANI, I. et al. **Improved Training of Wasserstein GANs**. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1704.00028>. Cited 2 times in pages 18 and 30.

25 IOFFE, S.; SZEGEDY, C. **Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift**. arXiv, 2015. Disponível em: <https://arxiv.org/abs/1502.03167>. Cited in page 18.

26 BA, J. L.; KIROS, J. R.; HINTON, G. E. **Layer Normalization**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1607.06450>. Cited 2 times in pages 18 and 19.

27 SALIMANS, T.; KINGMA, D. P. **Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks**. 2016. Cited in page 19.

28 MIYATO, T. et al. **Spectral Normalization for Generative Adversarial Networks**. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1802.05957>. Cited in page 19.

29 SALIMANS, T. et al. **Improved Techniques for Training GANs**. 2016. Cited in page 19.

30 RADFORD, A.; METZ, L.; CHINTALA, S. **Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks**. arXiv, 2015. Disponível em: <https://arxiv.org/abs/1511.06434>. Cited in page 19.

31 DENTON, E. et al. **Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks**. 2015. Cited in page 20.

32 KARRAS, T. et al. **Progressive Growing of GANs for Improved Quality, Stability, and Variation**. 2018. Cited 2 times in pages 20 and 21.

33 VASWANI, A. et al. **Attention Is All You Need**. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1706.03762>. Cited in page 20.

34 ZHANG, H. et al. **Self-Attention Generative Adversarial Networks**. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1805.08318>. Cited in page 20.

35 DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. **2009 IEEE conference on computer vision and pattern recognition**. [S.l.], 2009. p. 248–255. Cited in page 21.

36 HUANG, X.; BELONGIE, S. **Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization**. 2017. Cited in page 21.

37 DUMOULIN, V.; SHLENS, J.; KUDLUR, M. **A Learned Representation For Artistic Style**. 2017. Cited in page 21.

38 LI, Y. et al. **Universal Style Transfer via Feature Transforms**. 2017. Cited in page 21.

39 KARRAS, T.; LAINE, S.; AILA, T. **A Style-Based Generator Architecture for Generative Adversarial Networks**. Disponível em: <https://arxiv.org/abs/1812.04948>. Cited in page 21.

40 LAINE, S. **Feature-Based Metrics for Exploring the Latent Space of Generative Models**. 2018. Disponível em: <https://openreview.net/forum?id=BJslDBkwG>. Cited in page 21.

41 STYLEGAN3 pretrained models: Nvidia NGC. Disponível em: <https://catalog.ngc.nvidia.com/orgs/nvidia/teams/research/models/stylegan3>. Cited 4 times in pages 21, 22, 39, and 40.

42 KARRAS, T. et al. **Alias-Free Generative Adversarial Networks**. 2021. Cited in page 21.

43 HEUSEL, M. et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. arXiv, 2017. Disponível em: <https://arxiv.org/abs/1706.08500>. Cited 3 times in pages 21, 27, and 40.

44 ROMBACH, R. et al. **High-Resolution Image Synthesis with Latent Diffusion Models**. 2022. Cited in page 21.

45 DHARIWAL, P.; NICHOL, A. **Diffusion Models Beat GANs on Image Synthesis**. arXiv, 2021. Disponível em: <https://arxiv.org/abs/2105.05233>. Cited in page 21.

46 HO, J.; JAIN, A.; ABBEEL, P. **Denoising Diffusion Probabilistic Models**. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2006.11239>. Cited in page 21.

47 DEVLIN, J. et al. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**. 2019. Cited in page 21.

48 RADFORD, A. et al. Learning transferable visual models from natural language supervision. **CoRR**, abs/2103.00020, 2021. Disponível em: <https://arxiv.org/abs/2103.00020>. Cited 2 times in pages 21 and 24.

49 SAUER, A.; SCHWARZ, K.; GEIGER, A. **StyleGAN-XL: Scaling StyleGAN to Large Diverse Datasets**. 2022. Cited in page 22.

50 TAO, M. et al. **GALIP: Generative Adversarial CLIPs for Text-to-Image Synthesis**. 2023. Cited in page 22.

51 KANG, M. et al. Scaling up gans for text-to-image synthesis. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2023. Cited 3 times in pages 22, 25, and 26.

52 SAUER, A. et al. **Projected GANs Converge Faster**. 2021. Cited 2 times in pages 22 and 31.

53 REED, S. et al. **Learning Deep Representations of Fine-grained Visual Descriptions**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1605.05395>. Cited 2 times in pages 23 and 24.

54 MIKOLOV, T. et al. **Efficient Estimation of Word Representations in Vector Space**. arXiv, 2013. Disponível em: <https://arxiv.org/abs/1301.3781>. Cited in page 23.

55 XU, T. et al. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. **CoRR**, abs/1711.10485, 2017. Disponível em: <http://arxiv.org/abs/1711.10485>. Cited 5 times in pages 23, 25, 28, 35, and 44.

56 SZEGEDY, C. et al. **Rethinking the Inception Architecture for Computer Vision**. arXiv, 2015. Disponível em: <https://arxiv.org/abs/1512.00567>. Cited 2 times in pages 23 and 26.

57 YE, H. et al. Improving text-to-image synthesis using contrastive learning. **CoRR**, abs/2107.02423, 2021. Disponível em: <https://arxiv.org/abs/2107.02423>. Cited 2 times in pages 23 and 35.

58 ZHU, M. et al. DM-GAN: dynamic memory generative adversarial networks for text-to-image synthesis. **CoRR**, abs/1904.01310, 2019. Disponível em: <http://arxiv.org/abs/1904.01310>. Cited 3 times in pages 23, 35, and 44.

59 VASWANI, A. et al. **Attention Is All You Need**. 2017. Cited in page 24.

60 DOSOVITSKIY, A. et al. **An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale**. 2021. Cited in page 24.

61 STAP, D. et al. **Conditional Image Generation and Manipulation for User-Specified Content**. 2020. Cited in page 25.

62 PEREIRA, V. G.; WEHRMANN, J. Teaching stylegan to read: Improving text-to-image synthesis with u2c transfer learning. In: **33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022**. BMVA Press, 2022. Disponível em: <https://bmvc2022.mpi-inf.mpg.de/0512.pdf>. Cited in page 25.

63 YUAN, M.; PENG, Y. Bridge-gan: Interpretable representation learning for text-to-image synthesis. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 30, n. 11, p. 4258–4268, 2020. Cited in page 25.

64 ZHOU, Y. et al. Lafite: Towards language-free training for text-to-image generation. **arXiv preprint arXiv:2111.13792**, 2021. Cited 3 times in pages 25, 42, and 44.

65 SCHUHMANN, C. et al. **LAION-5B: An open large-scale dataset for training next generation image-text models**. 2022. Cited in page 26.

66 SALIMANS, T. et al. **Improved Techniques for Training GANs**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1606.03498>. Cited 2 times in pages 26 and 27.

67 BORJI, A. **Pros and Cons of GAN Evaluation Measures**. arXiv, 2018. Disponível em: <https://arxiv.org/abs/1802.03446>. Cited 2 times in pages 26 and 27.

68 BIńKOWSKI, M. et al. Demystifying MMD GANs. In: **International Conference on Learning Representations**. [s.n.], 2018. Disponível em: <https://openreview.net/forum?id=r1lUOzWCW>. Cited 3 times in pages 27, 28, and 40.

69 ZHANG, H. et al. **Cross-Modal Contrastive Learning for Text-to-Image Generation**. 2022. Cited 3 times in pages 28, 37, and 47.

70 KIROS, R.; SALAKHUTDINOV, R.; ZEMEL, R. S. **Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models**. 2014. Cited in page 29.

71 HINZ, T.; HEINRICH, S.; WERMTER, S. Semantic object accuracy for generative text-to-image synthesis. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Institute of Electrical and Electronics Engineers (IEEE), v. 44, n. 3, p. 1552–1565, mar 2022. Disponível em: <https://doi.org/10.1109%2Ftpami.2020.3021209>. Cited in page 29.

72 REDMON, J.; FARHADI, A. **YOLOv3: An Incremental Improvement**. 2018. Cited in page 29.

73 LIN, T. et al. Microsoft COCO: common objects in context. **CoRR**, abs/1405.0312, 2014. Disponível em: <http://arxiv.org/abs/1405.0312>. Cited in page 29.

74 HONG, S. et al. **Inferring Semantic Layout for Hierarchical Text-to-Image Synthesis**. 2018. Cited in page 29.

75 PAPINENI, K. et al. Bleu: a method for automatic evaluation of machine translation. In: **Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics**. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, 2002. p. 311–318. Disponível em: <https://aclanthology.org/P02-1040>. Cited in page 29.

76 BANERJEE, S.; LAVIE, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: **Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization**. Ann Arbor, Michigan: Association for Computational Linguistics, 2005. p. 65–72. Disponível em: <https://aclanthology.org/W05-0909>. Cited in page 29.

77 VEDANTAM, R.; ZITNICK, C. L.; PARIKH, D. **CIDEr: Consensus-based Image Description Evaluation**. 2015. Cited in page 29.

78 ZHUANG, F. et al. A comprehensive survey on transfer learning. **CoRR**, abs/1911.02685, 2019. Disponível em: <http://arxiv.org/abs/1911.02685>. Cited in page 30.

79 HOSNA, A. et al. Transfer learning: a friendly introduction. **Journal of Big Data**, v. 9, 10 2022. Cited in page 30.

80 IMAN, M.; ARABNIA, H. R.; RASHEED, K. A review of deep transfer learning and recent advancements. **Technologies**, MDPI AG, v. 11, n. 2, p. 40, mar 2023. Disponível em: <https://doi.org/10.3390%2Ftechnologies11020040>. Cited in page 30.

81 WANG, Y. et al. Transferring gans: generating images from limited data. **CoRR**, abs/1805.01677, 2018. Disponível em: <http://arxiv.org/abs/1805.01677>. Cited in page 30.

82 NOGUCHI, A.; HARADA, T. Image generation from small datasets via batch statistics adaptation. **CoRR**, abs/1904.01774, 2019. Disponível em: <http://arxiv.org/abs/1904.01774>. Cited in page 30.

83 WANG, Y. et al. Minegan: effective knowledge transfer from gans to target domains with few images. **CoRR**, abs/1912.05270, 2019. Disponível em: <http://arxiv.org/abs/1912.05270>. Cited in page 30.

84 MO, S.; CHO, M.; SHIN, J. **Freeze the Discriminator: a Simple Baseline for Fine-Tuning GANs**. arXiv, 2020. Disponível em: <https://arxiv.org/abs/2002.10964>. Cited in page 30.

85 ZHAO, M.; CONG, Y.; CARIN, L. On leveraging pretrained GANs for generation with limited data. In: III, H. D.; SINGH, A. (Ed.). **Proceedings of the 37th International Conference on Machine Learning**. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 11340–11351. Disponível em: <https://proceedings.mlr.press/v119/zhao20a.html>. Cited in page 30.

86 HAM, H.; JUN, T. J.; KIM, D. **Unbalanced GANs: Pre-training the Generator of Generative Adversarial Network using Variational Autoencoder**. 2020. Cited in page 30.

87 WANG, Y. et al. Distilling GANs with style-mixed triplets for x2i translation with limited data. In: **International Conference on Learning Representations**. [s.n.], 2022. Disponível em: <https://openreview.net/forum?id=QjOQkpzKbNk>. Cited in page 31.

88 PARK, T. et al. Semantic image synthesis with spatially-adaptive normalization. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2019. Cited in page 31.

89 CHOI, Y. et al. Stargan v2: Diverse image synthesis for multiple domains. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. Cited in page 31.

90 LIU, B. et al. **Towards Faster and Stabilized GAN Training for High-fidelity Few-shot Image Synthesis**. 2021. Cited in page 31.

91 LEE, K.-H. et al. Stacked cross attention for image-text matching. In: **Proceedings of the European conference on computer vision (ECCV)**. [S.l.: s.n.], 2018. p. 201–216. Cited in page 35.

92 WEHRMANN, J.; KOLLING, C.; BARROS, R. C. Adaptive cross-modal embeddings for image-text alignment. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2020. v. 34, n. 07, p. 12313–12320. Cited in page 35.

93 GITHUB - NVlabs/metfaces-dataset. <https://github.com/NVlabs/metfaces-dataset>. (Accessed on 07/02/2022). Cited in page 36.

94 STARGAN-V2/README.MD at master · clovaai/stargan-v2 · GitHub. <https://github.com/clovaai/stargan-v2/blob/master/README.md#animal-faces-hq-dataset-afhq>. (Accessed on 07/02/2022). Cited in page 36.

95 FROLOV, S. et al. Adversarial text-to-image synthesis: A review. **Neural Networks**, Elsevier BV, v. 144, p. 187–209, dec 2021. Disponível em: <https://doi.org/10.1016%2Fj.neunet.2021.07.019>. Cited 2 times in pages 37 and 47.

96 YU, F. et al. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. **CoRR**, abs/1506.03365, 2015. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1506.html#YuZSSX15>. Cited 2 times in pages 39 and 40.

97 YANG, S. et al. **From Facial Parts Responses to Face Detection: A Deep Learning Approach**. 2015. Cited 2 times in pages 39 and 40.

98 NVLABS. **Stylegan3/configs.md at main · nvlabs/stylegan3**. 2021. Disponível em: <https://github.com/NVlabs/stylegan3/blob/main/docs/configs.md>. Cited in page 40.

99 SALIMANS, T. et al. **Improved Techniques for Training GANs**. arXiv, 2016. Disponível em: <https://arxiv.org/abs/1606.03498>. Cited in page 40.

100 YE, S.; LIU, F.; TAN, M. **Recurrent Affine Transformation for Text-to-image Synthesis**. arXiv, 2022. Disponível em: <https://arxiv.org/abs/2204.10482>. Cited 2 times in pages 42 and 44.

101 LI, B. et al. Lightweight generative adversarial networks for text-guided image manipulation. **Advances in Neural Information Processing Systems**, v. 33, 2020. Cited 2 times in pages 42 and 44.