HyperGAN-CLIP: A Versatile Framework for CLIP-Guided Image Synthesis and Editing using Hypernetworks

by

Abdul Basit Anees

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Computer Science and Engineering

July 5, 2023

HyperGAN-CLIP: A Versatile Framework for CLIP-Guided Image Synthesis and Editing using Hypernetworks

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Abdul Basit Anees

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Prof. Dr. Deniz Yuret

_____

Prof. Dr. Yücel Yemez

_____

Prof. Dr. Hazım Kemal Ekenel

Date: _____

Dedicated to my family and friends who have always believed in me.

ABSTRACT

HyperGAN-CLIP: A Versatile Framework for CLIP-Guided Image Synthesis and Editing using Hypernetworks

Abdul Basit Anees

Master of Science in Computer Science and Engineering

July 5, 2023

Generative Adversarial Networks, particularly StyleGAN and its variants, have shown exceptional capability in generating highly realistic images. However, training these models remains challenging in domains where data is scarce, as it typically requires large datasets. In this thesis work, we introduce a versatile framework that enhances the capabilities of a pre-trained StyleGAN for various tasks, including domain adaptation, reference-guided image synthesis, and text-guided image manipulation even when only a small number of training sample are available. We achieve this by integrating the CLIP space into the generator of StyleGAN using hypernetworks. These hypernetworks introduce dynamic adaptability, enabling the pre-trained StyleGAN to be effectively applied to specific domains described by either a reference image or a textual description. To further improve the alignment between the synthesized images and the target domain, we introduce a CLIP-guided discriminator, ensuring the generation of high-quality images. Notably, our approach shows remarkable flexibility and scalability, enabling text-guided image manipulation with text-free training and seamless style transfer between two images. Through extensive qualitative and quantitative experiments, we validate the robustness and effectiveness of our approach, surpassing existing methods in terms of performance.

ÖZETÇE

Yüksek Lisans Tez Başlığı
Abdul Basit Anees
Bilgisayar Bilimleri ve Mühendisliği, Yüksek Lisans
5 Temmuz 2023

Generative Adversarial Networks, özellikle StyleGAN ve farklı türkri, son derece gerçekçi görüntüler oluşturmada olağanüstü bir yetenek göstermiştir. Bununla birlikte, bu modellerin eğitimi, genellikle büyük veri kümeleri gerektirdiğinden, verilerin az olduğu alanlarda zor olmaya devam etmektedir. Bu tez çalışmasında, yalnızca az sayıda eğitim örneği kullanıldığında bile alan uyarlaması, referans kılavuzlu görüntü sentezi ve metin kılavuzlu görüntü manipülasyonu dahil olmak üzere çeşitli görevler için önceden eğitilmiş bir StyleGAN'ın yeteneklerini artıran çok yönlü bir çerçeve yöntemi sunuyoruz. Bunu, hiper ağlar kullanarak CLIP alanını StyleGAN üretecine entegre ederek başarmaktayiz. Bu hiper ağlar, önceden eğitilmiş StyleGAN'ın bir referans görüntü veya metinsel bir açıklama ile tanımlanan belirli alanlara etkili bir şekilde uygulanmasını sağlayan dinamik uyarlanabilirlik sağlar. Sentezlenmiş görüntüler ile hedef alan arasındaki uyumu daha da iyileştirmek için, yüksek kaliteli görüntülerin üretilmesini sağlayan CLIP kılavuzlu bir ayırıcı sunuyoruz. Özellikle, yaklaşımımız kayda değer bir esneklik ve ölçeklenebilirlik göstererek, metinsiz eğitim ve iki görüntü arasında kesintisiz stil aktarımıyla metin kılavuzluğunda görüntü manipülasyonuna olanak tanır. Kapsamlı niteliksel ve niceliksel deneyler yoluyla, performans açısından mevcut yöntemleri geride bırakarak yaklaşımımızın sağlamlığını ve etkililiğini ortaya koymekteyiz.

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude to my advisors Prof. Deniz Yuret and Prof. Aykut Erdem for their thorough support and guidance during the course of my studies and research. I would also like to thank Prof. Erkut Erdem and Dr. Duygu Ceylan for their valuable feedback and help, which is pivotal to the completion of this thesis work. I am grateful to Prof. Deniz Yuret, Prof. Yücel Yemez and Prof. Hazım Kemal Ekenel for participating in my thesis committee. I acknowledge the financial support provided by KUIS AI Center as AI Fellowship for my master's studies.

I would also like to thank my friends at the KUIS AI lab for their friendship, motivation and support throughout. Among them, I would also like to thank Canberk Baykal for his valuable support as a group member of this project. Finally, I want to thank my family for their unconditional love and support; my mother, father, brother, sister and all those who always believed in me and motivated me to push myself beyond my limits.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

GAN	Generative Adversarial Network
CLIP	Contrastive Language-Image Pretraining
FID	Frechet Inception DIstance
e4e	encoder4editing
PCA	Principal Component Analysis
FFHQ	Flickr-Faces-HQ
CUB	Caltech-UCSD Birds
AMA	Attribute Manipulation Accuracy
CMP	CLIP Manipulative Precision

Chapter 1

INTRODUCTION

1.1 Motivation

Recent advances in Generative Adversarial Networks (GANs) [Goodfellow et al., 2014b] have made it possible to synthesize incredibly realistic-looking images. With architectural innovations such as progressive growth and style-based generators such as StyleGAN [Karras et al., 2019], one can train state-of-the-art GAN models on high-resolution datasets while controlling the style of the generated images through a semantically coherent latent space. However, training GANs often necessitates large amounts of data, which can be challenging or expensive to acquire in certain domains, making their applications limited to data-rich domains. To alleviate this problem, existing approaches for domain adaptation propose fine-tuning pre-trained generator networks to generate images in a target domain using a limited number of training samples [Ojha et al., 2021]. Nevertheless, when updating the generator, these methods frequently encounter a trade-off between capturing the diverse characteristics of the target domain and preserving the ability to generate high-quality images from the source domain. Also, fine-tuning for multiple target domains necessitates training a separate model for each of those domains.

Alongside the progress in generative learning, interesting breakthroughs have been observed in multimodal learning as well. Especially, the recently proposed Contrastive Language-Image Pre-training (CLIP) model [Radford et al., 2021] provides a joint common embedding for images and text. It is extremely rich in semantics especially since it is grounded with language as well. Hence, it makes great sense to use those learnt semantics for generative modelling tasks. Moreover, the joint image-text embedding space opens doors for text-based generation as well.

In tasks such as guided image generation and manipulation, where a reference text/ image is used for conditional image generation process within the pre-trained domain, the scarcity of paired training data can also restrict the capabilities of GANs. Some methodologies [Gal et al., 2022, Zhu et al., 2022] suggest leveraging the multi-modal CLIP embeddings, which project images and texts into a joint semantic space. However, such approaches are often limited to attributes seen during training [Wei et al., 2022, Lyu et al., 2023, Hidden, 2023] and face difficulties when conditioning on a guide image/ text that lies outside the distribution of the training set. Alternatively, some methods [Patashnik et al., 2021, Xia et al., 2021, Chefer et al., 2022] perform per-edit optimization that generalizes well, but is highly costly at inference time and is not scale-able.

Our work presents a general-purpose flexible framework that facilitates adapting pre-trained GAN models to multiple new domains using only a single example per each target domain while maintaining high image quality. Moreover, it also demonstrates a way to perform accurate reference guided image generation using the same framework.

1.2 Problem Definition

Domain adaptation for GANs involves adapting a pre-trained model to capture the distribution of a new image domain with limited training data, typically just a few samples, by utilizing the structure and semantics learnt from the pre-trained domain. If for a certain latent code, the source generator generates an image of a person, the objective of domain adaptation is to update the network such that the same latent code generates the image of the same man in terms of identity while only adding the style of the target domain. However, directly fine-tuning pre-trained GANs on the target domain with only the GAN loss can lead to overfitting and mode collapse. An example of that would be that all the generated images after fine-tuning look close the target domain images in the training set.

To address these challenges, it is required that some additional structural or semantic consistency be enforced during the fine-tuning process so that the identity

and structure from the source domain is preserved and only the target style is changed.

Moreover, an effective training strategy is required that ensures that the high fidelity generations are still maintained and the fine-tuned generator correctly represents the target domain.

Additionally, the process of fine-tuning has a limitation of specializing to only a single target domain at a time. Therefore, separate models need to be trained for each target domain costing both in terms of space for saving the models and time for training new models for each new domain.

Our work presents a general-purpose flexible framework that facilitates adapting pre-trained GAN models to multiple domains using only a single example per each target domain. Central to our method is a conditional hypernetwork that predicts modulation weights for the pre-trained generator, utilizing target domain embeddings obtained from an example. Specifically, our formulation enables dynamic adaptation of the generator's weights, effectively shifting its domain based on modulation coefficients while preserving the original capabilities of the generator. We represent the target embeddings by using the CLIP model to enable multi-modal domain adaptation that can be driven by either a target image or text caption. We note that we achieve this without any prior exposure to text data during pre-training. Moreover, we utilize a conditional discriminator during training that is conditioned on the CLIP embeddings. This enables the generation of high-quality and high-fidelity images that possess the characteristics of the target domain. Our hypernetwork has a lightweight architecture enabling on-the-fly domain adaption in a computationally efficient manner.

1.3 Major Contributions

In summary, we highlight our contributions as follows:

- We propose a conditional hypernetwork for one-shot domain adaptation using CLIP embeddings. It supports synthesis using both images and text due to the multimodal nature of CLIP.

- Our framework is a single model that supports various synthesis and editing applications without separate models, including generation in new target domains, style transfer between images, and text-guided manipulation, made possible through the use of CLIP embeddings

- For the task of text-guided image synthesis/ editing, our framework does not require any text labels during training.

- Our model supports the adaptation to any number of target domains without increasing the model size.

- We perform extensive evaluations on multiple datasets and confirm the superiority of our framework over existing approaches, demonstrating its effectiveness and robustness both qualitatively and quantitatively.

- We show that our approach is able to retain the source identity the best compared to other approaches due to the residual feature injection (Section 3.2). Instead of directly tuning the generator via the conditional hypernetwork, we consider a copy of the generator network and use that to generate missing features corresponding to the target domain using CLIP embeddings, which are then injected into the original frozen generator network.

1.4 Organization of the Thesis

The structure of this study is organized as follows:

- Chapter 2 reviews the prior work that is related to our work.

- Chapter 3 describes on the architectural details of our framework as well as the important loss functions that we use to adapt the GAN.

- Chapter 4 discusses the different experimental settings or tasks that we test our framework on. We also highlight the impact of each component of our approach through comprehensive ablations.

- Chapter 5 concludes the thesis work by a short summary and outlines possible future research directions.

Chapter 2

RELATED WORK

2.1 State-of-the-art in GANs

Generative Adversarial Networks (GAN) [Goodfellow et al., 2014b] consist of a generator network and a discriminator network that are competing against each other in a min-max game. The generator maps a random noise vector $\mathcal{Z}$ to credible data samples from the data distribution. The discriminator is a classification model that distinguishes between the real data samples coming from the real data distribution and the fake data samples that are produced by the generator.

In recent years, the field of image synthesis and editing has witnessed remarkable progress through the use of generative adversarial networks (GANs)[Goodfellow et al., 2014a]. This progress has been driven by innovative combinations of architecture and training techniques, resulting in the generation of highly realistic images. Some notable examples of these techniques include PGGAN [Karras et al., 2018], which employs progressive growth of the image resolution through cascaded generators to add more and more fine-scale details to the generated images, and BigGAN [Brock et al., 2019], which scales up class-conditioned image synthesis by increasing the batch sizes and the width in each layer and introducing residual connection along with the so-called truncation trick to generate high-quality images.

Motivated by the studies on style transfer [Gatys et al., 2015], StyleGAN proposes a novel generator that first maps the input latent variable z to an intermediate latent space $\mathcal{W}$ using a mapping network. It introduces style mixing by means of Adaptive Instance Normalization (AdaIN) [Huang and Belongie, 2017], where latent codes from this intermediate $\mathcal{W}$ space control the scale and bias parameters for the normalization of feature maps. This unique architecture is capable of generating highly realistic images while also allowing control over the coarse, medium, and fine-

scaled style of the synthesized images by means of latent code fed into each stage. StyleGAN2 [Karras et al., 2020] proposes several modifications to StyleGAN to improve image quality and realism. To remove the water droplet-like artifacts, Adaptive Instance Normalization is replaced by weight demodulation. Additionally, by adding more generator regularization and improving the progressive growth regime, StyleGAN2 further boosts the performance of the original StyleGAN. More recently, StyleGAN3 [Karras et al., 2021] tackles the texture sticking problem and further reduces the visible artifacts by making the underlying architecture fully equivariant to translation and rotation. Among these alternative, StyleGAN2 is used for this study for a fair and accurate comparison against the competing approaches.

Other recent models such as StyleSwin [Zhang et al., 2022] and GANformer [Hudson and Zitnick, 2021] use transformers or bipartite structures to generate high-quality images containing multiple objects.

Latent Space Manipulation StyleGAN is renowned for its ability to offer a semantically meaningful latent space, allowing users to manipulate attributes in synthesized images by navigating semantic directions within this space. To edit a real image using StyleGAN, a common approach is to perform GAN inversion, which involves mapping an input image into the latent space such that that latent code produces the same image when fed into the generator. GAN inversion can be achieved through various techniques, such as direct optimization [Creswell and Bharath, 2019, Abdal et al., 2019, Abdal et al., 2020, Tewari et al., 2020] (where an initial latent code is optimized until the generator starts to produce the desired image), learning-based methods [Zhu et al., 2020, Alaluf et al., 2021, Bau et al., 2019a, Richardson et al., 2021, Tov et al., 2021, Bai et al., 2022] (involves training an image encoder to map image to latent space), or hybrid methods [Zhu et al., 2016, Bau et al., 2019b] (combination of former both). Once the image is inverted into the latent space, users can explore the manifold to discover editing directions that hold semantic significance.

Notably, extensive research has demonstrated that these editing directions can be discovered in an unsupervised manner [Voynov and Babenko, 2020, Shen and

Zhou, 2021]. InterFaceGAN [Shen et al., 2020b] finds hyperplanes that separate the plane in two based on a binary attribute and consequently, finds latent directions to manipulate real images. GANSpace [Härkönen et al., 2020] applies Principal Component Analysis in the latent or the feature space to find editing directions for different attributes. Other studies have focused on leveraging image-level attributes to explicitly identify these directions [Shen et al., 2020a, Abdal et al., 2021, Wu et al., 2021].

There are different latent space in the StyleGAN generator to consider for madnipulation. The latent mapper transforms the $\mathcal{Z}$ space latent code drawn from a Normal distribution to an intermediate latent space $\mathcal{W}$. Then, the $\mathcal{W}$ space code is input at different stages in the StyleGAN2 generator. It is further transformed to the $\mathcal{S}$ space by an affine transformation at each stage. Another space is the $\mathcal{W}+$ space, where we may choose to input different $\mathbf{W}$ for each stage. Different works use different spaces. For example, some works use the $\mathcal{S}$ space for manipulation such as StyleCLIP-GD [Patashnik et al., 2021] and StyleMC [Kocasari et al., 2021]. Others like StyleCLIP-LO, StyleCLIP-LM [Patashnik et al., 2021], HairCLIP [Wei et al., 2022] utilize the extended intermediate $mathcalW+$ space.

2.2 Domain Adaptation for GANs

Few-shot domain adaptation for GANs involves adapting a pre-trained model to capture the distribution of a new image domain with limited training data, typically just a few samples. Directly fine-tuning pre-trained GANs on the target domain can lead to overfitting and mode collapse. To address this challenge, recent works have proposed various strategies. [Ojha et al., 2021] introduce a cross-domain distance consistency loss to preserve relative similarities and differences from the source domain, effectively transferring diversity to the target domain. RSSA [Xiao et al., 2022] uses a cross-domain spatial structural consistency loss to compress the latent space to a subspace aligned with the target domain. StyleGAN-NADA [Gal et al., 2022] leverages the joint embedding space of the CLIP model [Radford et al., 2021], guiding the adaptation with a directional CLIP loss. Mind the Gap [Zhu et al.,

2022] incorporates regularizers and additional directional losses to mitigate overfitting and mode collapse. For adapting to multiple domains, DynaGAN [Kim et al., 2022b] employs hypernetworks as adaptation modules to modulate the weights of the pre-trained generator based on the one-hot vectors of the target domains. HyperDomainNet [Alanov et al., 2023] also utilize hypernetworks that predict additional weight modulation parameters along with some regularizers and an additional CLIP directional loss for multi-domain adaptation. Domain expansion [Nitzan et al., 2023] repurposes latent directions that have minimal perceptible effects on the output images, assigning them to represent the target domains. Similar to DynaGAN, we use a hypernetwork to modulate the weights of a pre-trained StyleGAN2 generator, also allowing for adapting multiple domains with a single network model. However, our hypernetwork is conditioned on the CLIP embeddings of the target domains, enabling our approach to generalize to unseen target domains during training. We further utilize a novel CLIP-conditioned discriminator to ensure better sample quality.

2.3 Reference-Guided Image Synthesis

Reference-guided image synthesis aims to generate images by combining the content of an input image with the style of a reference image. While earlier works on neural style transfer [Gatys et al., 2015] have achieved success in transferring arbitrary or multiple styles to images, they often suffer from artifacts due to neglecting local semantic styles. To overcome this limitation, WCT^2[Yoo et al., 2019] introduces wavelet-corrected transfer, preserving both structural information and local feature statistics in the VGG feature space. DeepFaceEditing [Chen et al., 2021] employs local disentanglement and global fusion modules to better separate the geometry and appearance of synthesized images. BlendGAN [Liu et al., 2021a] takes a self-supervised approach, training a style encoder and incorporating a weighted blending module for seamless fusion of face and style representations. TargetCLIP [Chefer et al., 2022] explores the StyleGAN2 latent space to find an essence vector that represents the desired editing direction based on a reference image, optimizing it

to maximize similarity between the synthesized image and the target image in the CLIP space. NeRFFaceEditing [Jiang et al., 2022] introduces appearance and geometry decoders within a pre-trained tri-plane-based neural radiance field, decoupling appearance and geometry with an AdaIN-based method. In contrast to previous approaches, our proposed HyperGAN-CLIP model empowers the CLIP embeddings of the target image to control the modulation weights, while simultaneously decoding the latent representation of the source image using the StyleGAN2 generator.

2.4 Text-Guided Image Manipulation

Text-guided manipulation involves modifying an image based on a textual description while preserving its structure and incorporating the desired attributes described in the text. Recent advancements in this field have leveraged multi-modal approaches, such as CLIP [Radford et al., 2021], which establish a shared latent space for images and text. These approaches enable precise and intuitive editing capabilities guided by text. StyleCLIP-LO [Patashnik et al., 2021] optimizes the latent code to generate a target image by minimizing the distance between image-text pairs in the CLIP space. StyleCLIP-LM [Patashnik et al., 2021] predicts residual latent codes based on the CLIP similarity of attributes and output images. StyleCLIP-GD [Patashnik et al., 2021] maps text prompts to global directions in the original StyleGAN space, while StyleMC [Kocasari et al., 2021] learns global directions in the lower dimensional $\mathcal{S}$ space of StyleGAN2. These methods utilize CLIP to align generated images with desired textual descriptions. HairCLIP [Wei et al., 2022] modulates latent codes based on target hairstyle and hair color using reference images or text, optimizing similarity in the CLIP space. DeltaEdit [Lyu et al., 2023] trains latent mappers solely on images using semantically aligned Δ-CLIP space, enabling manipulations guided by reference textual descriptions or images. CLIPInverter [Hidden, 2023]conditions the inversion stage on textual descriptions, obtaining manipulation directions as residual latent codes through a CLIP-guided adapter module. In the context of diffusion-based synthesis methods, DiffusionCLIP [Kim et al., 2022a] converts input images to noise through forward diffusion

and fine-tunes the score function in the reverse diffusion process using CLIP similarity, while Plug-and-play [Tumanyan et al., 2022] injects image feature maps from latent diffusion model denoising into the image synthesis process guided by textual descriptions. Unlike these works, except DeltaEdit, our approach does not rely on textual data during training. By utilizing the Δ-CLIP space to guide our hypernetwork, we achieve zero-shot prediction of modulation weights and competitive text-driven editing results during inference.

2.5 Hypernetworks

Hypernetworks [Ha et al., 2017] refers to neural networks that are trained to predict or modulate the weights of another neural network called the primary network. This unique capability enables hypernetworks to enhance expressivity and generalizability in a wide range of applications, making the primary network to adapt itself to different contexts in a dynamic manner. Important examples include 3D modeling [Littwin and Wolf, 2019, Spurek et al., 2021], semantic segmentation [Yuval Nirkin, 2021], and continual learning [von Oswald et al., 2020]. In the context of GANs, hypernetworks have also emerged as a valuable tool for GAN inversion and domain adaptation. HyperInverter [Dinh et al., 2022] utilizes hypernetworks while predicting the parameters of the encoder network. HyperStyle [Alaluf et al., 2022], on the other hand, inverts a given image into the StyleGAN latent space by adapting the generator via hypernetworks, enabling the latent space to better represent out-of-domain images. Additionally, DynaGAN [Kim et al., 2022b] leverages hypernetworks for few-shot domain adaptation by dynamically modulating the weights of the StyleGAN model. Inspired by these studies, our approach capitalizes on the flexibility of hypernetworks to modulate the weights of StyleGAN based on CLIP by harnessing its multimodal nature across different modalities. This novel integration of hypernetworks and CLIP opens up exciting possibilities for image synthesis and editing, as will be discussed in the subsequent sections.

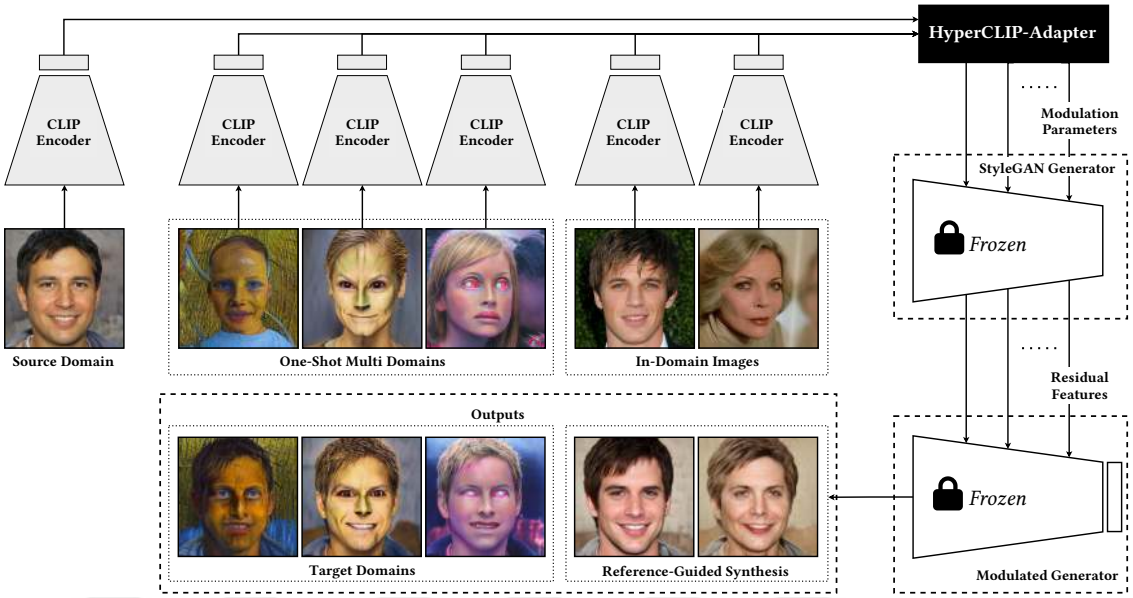

Figure 3.1: **Overview of HyperGAN-CLIP, our one-shot domain adaptation framework**. The framework leverages the HyperCLIP-Adapter, a hypernetwork module, to predict modulation weights for a pre-trained StyleGAN generator based on target domain information. By modulating the generator weights and combining modulated convolutional features with the original ones, it is capable of generating images from the target domain while preserving source domain information. Our framework also supports in-domain adaptation without modification and enables reference-guided image synthesis. Additionally, it has the potential for text-guided image manipulation, although not shown in the figure.

Chapter 3

APPROACH

In this section, we begin by presenting the problem definition and providing an overview of our one-shot domain-adaptation framework called HyperGAN-CLIP (Sec. 3.1). Subsequently, we delve into the detailed exploration of the main components of our proposed model. Specifically, we discuss our hypernetwork-based adapter layer (Sec. 3.2) and the Δ-CLIP embeddings (Sec. 3.3), which play crucial roles in our framework. We then describe the training loss functions, including

our CLIP-based conditional discriminator (Sec. 3.4). Finally, we examine how our framework can effectively address reference-guided image synthesis and text-guided image manipulation without requiring any modification of our model architecture (Sec. 3.5). For an overview of our HyperGAN-CLIP framework, please refer to Fig. 3.1.

3.1 Overview

Problem Formulation of Domain Adaptation Assume a GAN generator G_{source} pre-trained on a source domain $\mathcal{D}_{\text{source}}$ using large amount of data. The generator G_{source} generates an image $G_{\text{source}}(z)$ by mapping a noise vector z sampled from the latent space, $z \sim p(z) \subset \mathbb{R}^d$. In few-shot domain adaptation, the aim is to adapt G_{source} to a target domain $\mathcal{D}_{\text{target}}$ using only a few samples from the target domain, resulting in a new generator G_{target}. Here, the training set could even consist of a single image, which is referred to as one-shot domain adaptation. In the most simple way, the adaptation can be achieved by initializing the weights of G_{target} with the pretrained weights of G_{source} and then performing finetuning using the GAN training objective as below:

$$\mathcal{L}_G = -\mathbb{E}_{z\sim p(z)}\left[\log\left(D\left(G_{\text{target}}(z)\right)\right)\right] \tag{3.1}$$

$$\mathcal{L}_D = \mathbb{E}_{x\sim \mathcal{D}_{\text{target}}}\left[\log(1 - D(x)\right] + \mathbb{E}_{z\sim p(z)}\left[\log\left(D\left(G_{\text{target}}(z)\right)\right)\right] \tag{3.2}$$

where D represents the discriminator whose weights are transferred from the source domain and can also be finetuned. As mentioned before, this naive approach may suffer from mode collapse, leading the generator to generate (memorize) images from training data. Moreover, this is not a scalable solution, as a separate generator should be trained per each target domain.

Multiple Domain One-Shot Generator Adaptation To tackle the aforementioned issues, our model incorporates an adapter layer that predicts modulation weights for a pre-trained StyleGAN2 [Karras et al., 2020] generator network through a hypernetwork conditioned on the target domain. By updating the generator

weights, we effectively shift the generator's domain, enabling it to synthesize images from the target domain instead of its original pre-trained domain. To retain maximum information from the source domain, we propose to insert the features obtained with the adapted weights as a residual to the features extracted by the frozen weights at each stage of the network. This novel approach enables us to employ a single generator model capable of dynamically adapting to *multiple* target domains via our lightweight hypernetwork layer, while preserving its original generation capability. Furthermore, we address the challenging case of *one-shot* adaptation where each target domain is expressed with just a single image.

To obtain the target embeddings, we utilize the shared image and text embedding space provided by the CLIP model [Radford et al., 2021]. This makes our hypernetwork capable to seamlessly operate on both target images and text, even without prior exposure to textual data during training. Formally, given a StyleGAN2 generator G_{source} pretrained on a source domain and a set of N domain adaptation tasks, we aim at training a single generator $G_\star$ which is capable of modeling all the target domains $\cup_{i=1}^{N} \mathcal{D}_{\text{target}_i}$. We adapt $G_\star$ into a conditional generator where each target domain $\mathcal{D}_{\text{target}_i}$ is encoded by the CLIP embedding of the single training image, denoted by c_i. An image generated in the source domain, $x = G_{\text{source}}(z)$, can be translated into a target domain $\mathcal{D}_{\text{target}_i}$ by the mapping $G_\star(z, c_i)$ such that the same latent code z generates an image x_{target} which reflects the style or attributes of the target domain while preserving the identity of the subject from the source domain. Hence, we formulate domain adaptation as projecting the CLIP embeddings to the weight modulation parameters such that the weight modulation adapts the generator to the target domain.

3.2 HyperCLIP-Adapter

The architecture of StyleGAN2 generator consists of a mapping network and a synthesis network. The mapping network takes a random noise vector z sampled from the initial latent space (typically modelled with a multivariate normal distribution) as input and map the original latent z into the intermediate $\mathcal{W}$ space via multiple

fully connected layers. The corresponding latent code w is further transformed into what is known as the $\mathcal{S}$ space using learned affine transformations for each layer of the synthesis network [Wu et al., 2021]. The style vector s is then used to apply channel-wise scaling (modulation) to the convolutional weights ϕ_i at layer i followed by a demodulation operation to get the final weights θ_i, given as:

$$\theta_i = f(\phi_i, s_i) \tag{3.3}$$

Where $f = \text{Demod} \circ \text{Mod}\ (\cdot)$ denotes the composite function performing cascaded modulation and demodulation operations. Performing modulation in coarse layers is sufficient for generating in-domain variation. However, domain adaptation requires more fine-grained modulation of the network weights.

In our approach, we propose modulating StyleGAN2 weights further using target domain information extracted from the CLIP embedding. We manage this through the use of a CLIP conditioned hypernetwork, which learns a mapping directly from CLIP space to the space of convolutional weights. We call this hypernetwork as HyperCLIP-Adapter. In particular, it dynamically predicts domain-specific weight bias $\Delta\phi_i$ and channel-wise scale δ_i, as given below:

$$(\Delta\phi_i, \delta_i) = \text{HyperCLIP-Adapter}_i(c) \tag{3.4}$$

where c is the CLIP embedding of the training image representing the target domain, the index i refers to the i-th layer of the generator.

These modulation parameters are estimated via fully-connected layers performing affine transformations. Domain-specific channel-wise scale δ_i can be estimated as:

$$\delta_i = A_{\delta_i}(c) = w_{\delta_i} \cdot c + b_{\delta_i}. \tag{3.5}$$

To compute the domain-specific weight bias $\Delta\phi_i$ in an efficient manner, following [Hou and Kung, 2021, Kim et al., 2022b], we reparameterize it as $\Delta\phi_i = \gamma_i \otimes \Phi_i \otimes \psi_i$ by using rank-1 tensor decomposition, which significantly reduces the computation burden. Then, these rank-1 decomposed modulation parameters γ_i,

Φ_i, ψ_i can be estimated as:

$$\gamma_i = A_{\gamma_i}(c) = w_{\gamma_i} \cdot c + b_{\gamma_i}, \tag{3.6}$$

$$\Phi_i = A_{\Phi_i}(c) = w_{\Phi_i} \cdot c + b_{\Phi_i}, \tag{3.7}$$

$$\psi_i = A_{\psi_i}(c) = w_{\psi_i} \cdot c + b_{\psi_i}. \tag{3.8}$$

Subsequently, the modulated StyleGAN2 weights are given as:

$$\theta_i^* = \delta_i \cdot f(\phi_i + \Delta\phi_i, s_i). \tag{3.9}$$

One can use these weights to estimate features from which target domain images are generated. In fact, a similar type of weight modulation has been considered in prior work [Kim et al., 2022b]. However, we observe that directly modulating the weights introduces a shift in the identity when adapting the source generator to a target domain. Moreover, it causes the generator to overfit to the target image, especially in one-shot domain adaptation. To tackle this, we fuse source domain features with target domain features via feature injection.

More specifically, we keep the original weights for stage i, θ_i, and extract intermediate original features F_i from $F_{i-1}^{'}$ using:

$$F_i = F_{i-1}^{'} \circledast \theta_i + b_i \tag{3.10}$$

where $\circledast$ is the convolution operator and b_i denotes the layer bias. We then estimate target domain features using the modulated weights θ_i^* as follows:

$$F_i^* = F_{i-1} \circledast \theta_i^* + b_i, \tag{3.11}$$

and inject them into the original features by scaling them down so that the final features remain close to the original distribution even at the start of training, as given below:

$$F_i^{'} = F_i + \eta \cdot F_i^*. \tag{3.12}$$

with η denoting the scaling parameter.

In short, instead of directly updating the original generator network, we update the weights for a copy of the generator network, which generates missing features

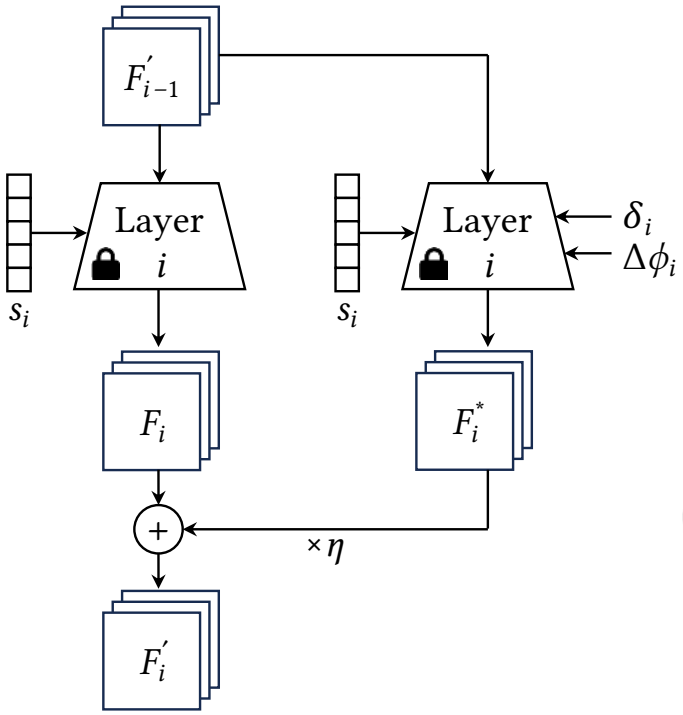

Figure 3.2: **Feature Injection**. We inject the domain specific residual features to the original features to keep source domain information to a certain extent while performing domain adaptation.

corresponding to the target domain using CLIP embeddings. Subsequently, these missing features are injected into the original frozen generator network through the proposed residual feature injection module.

It is important to emphasize that using CLIP embeddings to guide the modulation process via a hypernetwork introduces certain advantages. First, it enables information sharing between different domains. Additionally, it allows for the use of unseen target embeddings at inference time as well even when they are not seen during training as long as the target embedding lies within or close to the distribution of the training embeddings.

3.3 Δ-CLIP Embeddings

CLIP [Radford et al., 2021] has a shared image and text embedding space, it is also possible to interchangeably use image and text embeddings as target. However,

we observe that simply using raw CLIP text embeddings at inference changes the identity and degrades the image quality by a very noticeable factor (see Fig. 4.9). To solve this, we modify our training procedure such that instead of using the CLIP embedding as input, we use the difference of CLIP embeddings:

$$c_{\text{original}} = \text{CLIP}(x_{\text{source}}), \tag{3.13}$$

$$c_{\text{target}} = \text{CLIP}(x_{\text{target}}), \tag{3.14}$$

$$\Delta c = c_{\text{target}} - c_{\text{original}}. \tag{3.15}$$

In this way, we only feed the missing information about the target domain and remove any redundant information already present in the source domain image embedding. Moreover, this makes the input embeddings centered around zero making the training easier. A concurrent work [Lyu et al., 2023] has also shown that the difference of CLIP embeddings or the Δ-CLIP space provides a semantically more aligned shared space for image and text modalities.

3.4 Training Losses

We train our HyperGAN-CLIP framework by minimizing a multi-task loss $\mathcal{L}$, defined as:

$$\begin{aligned}\mathcal{L} = \lambda_1\mathcal{L}_{\text{CLIP}} + \lambda_2\mathcal{L}_{\text{CLIP-Across}} + \lambda_3\mathcal{L}_{\text{CLIP-Within}} + \lambda_4\mathcal{L}_{\text{cGAN}} \\ +\lambda_5\mathcal{L}_{\text{Contrastive}} + \lambda_6\mathcal{L}_{\text{ID}} + \lambda_7\mathcal{L}_{\text{L2}} + \lambda_8\mathcal{L}_{\text{LPIPS}}\end{aligned} \tag{3.16}$$

where λ_* depicts the corresponding regularization coefficients. We next detail each of these terms.

CLIP-based losses Given our one-shot target domain image x_{target}, our primary objective is to try to have similar semantics in the adapted domain images as in x_{target}. Assume z_{source} is the latent code which corresponds to the inversion of x_{target} in the source domain. In other words, z_{source} generates x_{fixed} with the frozen generator which is the corresponding source domain image of x_{target}. Our goal is to have the same latent code z_{source} generate an adapted image reconstruction x_{recon} with the

adapted generator. As we encode the target domain by the CLIP embedding of the corresponding training image, an obvious choice to enforce the consistency of adaption process is to maximize the pairwise CLIP similarity between the generated images and the target domain samples, defined as:

$$\mathcal{L}_{\text{CLIP}} = 1 - \langle c_{\text{recon}}, c_{\text{target}} \rangle \tag{3.17}$$

where c_{target} and c_{recon} express the CLIP embeddings of the target and the synthesized images, respectively, and $\langle \cdot, \cdot \rangle$ denotes the cosine similarity between them.

StyleGAN-NADA [Gal et al., 2022] showed that using such a global CLIP loss only suffers from a mode-collapse and does not preserve the original image content. Hence, the authors propose to use directional CLIP loss to capture the differences between the source and target domains and to increase the diversity. This idea is further explored in Mind the Gap [Zhu et al., 2022] to better preserve directions within and across domains. In our work, to increase coverage and faithfulness to target domains, we also employ these directional CLIP losses. Suppose that x_{sample} is an image generated by the frozen generator from a randomly sampled latent code z_{sample}. While training our model to adapt x_{sample} to the target domain $\mathcal{D}_t$ to generate x_{trained}, we cannot directly use a reconstruction-based loss since the identities of x_{sample} and x_{target} are different. However, in the semantic sense, we expect the same style to be added to x_{sample} to generate x_{trained} which was supposed to be added to x_{fixed} to generate x_{target}. The semantic shift between the two domains can be inherently quantified by vector arithmetic in the CLIP embedding space. More specifically, we require the direction from x_{sample} to x_{trained} to be the same as the direction from x_{fixed} to x_{target} in the CLIP embedding space. We do this by maximizing the cosine similarity between these directions, defined as follows:

$$\begin{aligned} \Delta c_{\text{sample}} &= \text{CLIP}(x_{\text{trained}}) - \text{CLIP}(x_{\text{sample}}) \\ \Delta c_{\text{fixed}} &= \text{CLIP}(x_{\text{target}}) - \text{CLIP}(x_{\text{fixed}}) \\ \mathcal{L}_{\text{CLIP-Across}} &= 1 - \langle \Delta c_{\text{sample}}, \Delta c_{\text{fixed}} \rangle \end{aligned} \tag{3.18}$$

During training, it is also likely that the adapted generator might change image characteristics that are not related to the domain gap, such as identity. Therefore,

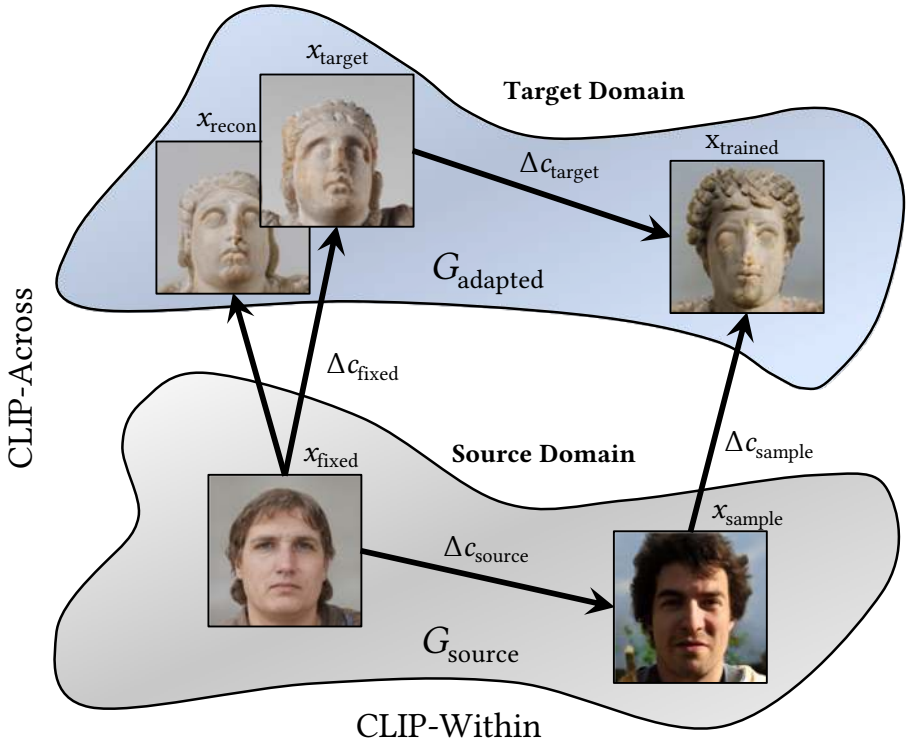

Figure 3.3: **Visualization of the directional CLIP losses for domain adaptation.** We encode the images generated by the original and the modulated generators, representing the source and target domains, in the CLIP space. The CLIP-Across loss, involving Δc_{sample} and Δc_{fixed}, captures the differences between the source and target domains. On the other hand, the CLIP-Within loss, computed using Δc_{source} and Δc_{target}, preserves the semantic information that is unrelated to the domain gap. These two loss components collectively enable the effective preservation of semantic information while capturing domain-specific variations.

we also regularize the adapted generator to preserve the semantic information that is not related to the domain gap. Specifically, we enforce the direction from x_{sample} to x_{fixed} in the source domain to be the same as the direction from x_{target} to x_{trained}, defined as:

$$\begin{aligned} \Delta c_{\text{source}} &= \text{CLIP}(x_{\text{fixed}}) - \text{CLIP}(x_{\text{sample}}) \\ \Delta c_{\text{target}} &= \text{CLIP}(x_{\text{target}}) - \text{CLIP}(x_{\text{trained}}) \\ \mathcal{L}_{\text{CLIP-Within}} &= 1 - \langle \Delta c_{\text{source}}, \Delta c_{\text{target}} \rangle \end{aligned} \tag{3.19}$$

For a better understanding of these losses and their roles, please refer to the graphical illustration provided in Fig. 3.3.

CLIP-conditioned discriminator loss Adapting a generator to a new domain usually deteriorates sample quality. We propose to alleviate this by additionally introducing an adversarial loss $\mathcal{L}_{\text{cGAN}}$, which is defined based on a conditioned discriminator. Following [Kumari et al., 2022], we employ a discriminator with a frozen vision transformer backbone taken from the CLIP model and train the outer most head layers. The discriminator acts like a learned loss function to measure the gap between source and target domain distributions. However, since we consider the one-shot setting, we only have a single target domain image. Therefore, we additionally use differentiable augmentation [Zhao et al., 2020] to deal with the data scarcity problem. Moreover, in order to further align the generated images with the target domains, we condition the discriminator on CLIP embeddings as proposed in [Miyato and Koyama, 2018]. The discriminator verifies whether an image and a corresponding target domain embedding belong together. This also leverages shared knowledge between domains, enabling the generator to adapt to multiple target domains while preventing mode collapse and retaining the diversity learned during pretraining. We observe that utilizing this discriminator also helps with faster training convergence.

Contrastive adaptation loss Given the inherent unique characteristics of each domain, it is crucial that the generated images from a target domain should exhibit semantic differences compared to those generated from other domains. This objective encourages the network to learn domain-specific transformations for individual target embeddings, resulting in a more disentangled representation for the various target domains. The contrastive-adaptation loss $\mathcal{L}_{\text{Contrastive}}$ proposed by [Kim et al., 2022b] promotes this following a contrastive learning objective. The key idea behind the contrastive-adaptation loss is to encourage similarity or dissimilarity between the synthesized image and a target training image, depending on whether they form positive or negative pairs. This implies that they belong to the same or different target domains, respectively. Supposing that x^k_{target} and x^k_{recon} denote a training image and an image synthesized for the k-th target domain $\mathcal{D}_{\text{target}_k}$, respectively, this loss is

formulated as:

$$\mathcal{L}_{\text{Contrastive}} = -\log \frac{\exp(l_{\text{pos}})}{\exp(l_{\text{pos}}) + \Sigma_j \mathbf{1}_{[j \neq k]} \exp(l^j_{\text{neg}})} \tag{3.20}$$

where l_{pos} and l^j_{neg} are cosine similarities of positive and negative pairs, defined as:

$$l_{\text{pos}} = \left\langle E_{\text{CLIP}}(x^k_{\text{target}}), E_{\text{CLIP}}(x^k_{\text{recon}}) \right\rangle \tag{3.21}$$

$$l^j_{\text{neg}} = \left\langle E_{\text{CLIP}}(\text{Aug}(x^j_{\text{target}})), E_{\text{CLIP}}(x^k_{\text{recon}}) \right\rangle \tag{3.22}$$

where E_{CLIP} is the CLIP encoder and $\text{Aug}(\cdot)$ denotes a horizontal-flip and color-jitter augmentation to improve training stability, as suggested in [Liu et al., 2021b].

Identity loss As an additional constraint to preserve the source identity in the target domain, we also employ an identity similarity loss that maximizes the cosine similarity between the image x_{sample} in the source domain and the image x_{trained} in the target domain.

$$\mathcal{L}_{\text{ID}} = 1 - \langle R(x_{\text{sample}})), R(x_{\text{trained}}) \rangle \,, \tag{3.23}$$

where $R(\cdot)$ denotes deep features extracted from the ArcFace [Deng et al., 2022] model pretrained for recognizing human faces.

Perceptual and Reconstruction losses In addition to the CLIP loss $\mathcal{L}_{\text{CLIP}}$, we further try aligning x_{recon} with x_{target} in the pixel and perceptual spaces based on the commonly used L2 and LPIPS losses, defined below:

$$\mathcal{L}_{\text{L2}} = \|x_{\text{target}} - x_{\text{recon}}\|_2 \tag{3.24}$$

$$\mathcal{L}_{\text{LPIPS}} = \|F(x_{\text{target}}) - F(x_{\text{recon}})\|_2 \tag{3.25}$$

where $F(\cdot)$ denotes deep features extracted from a pretrained AlexNet [Krizhevsky et al., 2012] model.

3.5 Extending HyperGAN-CLIP to Other Tasks

Building upon our proposed approach, we also explore two tasks related to in-domain generator adaptation. The first task is reference-guided image synthesis, which

involves generating images by seamlessly blending the content of one image with the style of another. The second task is text-guided image manipulation, where the aim is to modify the content of an image based on a given textual description that specifies the desired attribute changes.

To tackle these tasks, we take a unique approach by utilizing target images from the original source domain as opposed to considering target domains with drastically different visual characteristic, as explored in domain adaption. This choice allows our CLIP-conditioned hypernetwork, HyperCLIP-Adapter, to dynamically adjust the weights of the StyleGAN generator while considering in-domain data. By doing so, the generator can produce outputs that reflect the style of the provided target image while maintaining the facial identity of the input image. As will be demonstrated later, this formulation enables us to achieve more accurate and controlled image synthesis and editing within the same domain, resulting in high-quality and faithful results.

During the training process, we adopt a strategy where we randomly sample sets of source and target image pairs from the original data at each iteration, without relying on external data as used in domain adaptation experiments. This strategy allows us to cover a wide distribution of CLIP embeddings and enables the network to learn a strong relationship between the CLIP space and the StyleGAN image space. Furthermore, we slightly modify the CLIP-Across and CLIP-Within losses (see Fig. 3.4) since source and target images are from the same domain. Specifically, in the definition of the CLIP-Across loss, we use the mean StyleGAN image as the anchor image x_{fixed} instead of the inverted target image as in domain adaptation, and we only sample x_{target} and x_{sample} randomly during training. Similarly, in the CLIP-Within loss, we replace x_{target} with x_{recon} to better enforce the preservation of the identity and image content.

Furthermore, in our framework, the weight modulation is guided by the Δ-CLIP space through our HyperCLIP-Adapter module. This design choice allows our model trained for reference-guided image synthesis to be directly employed for text-guided image manipulation as well. In other words, our training procedure for reference-

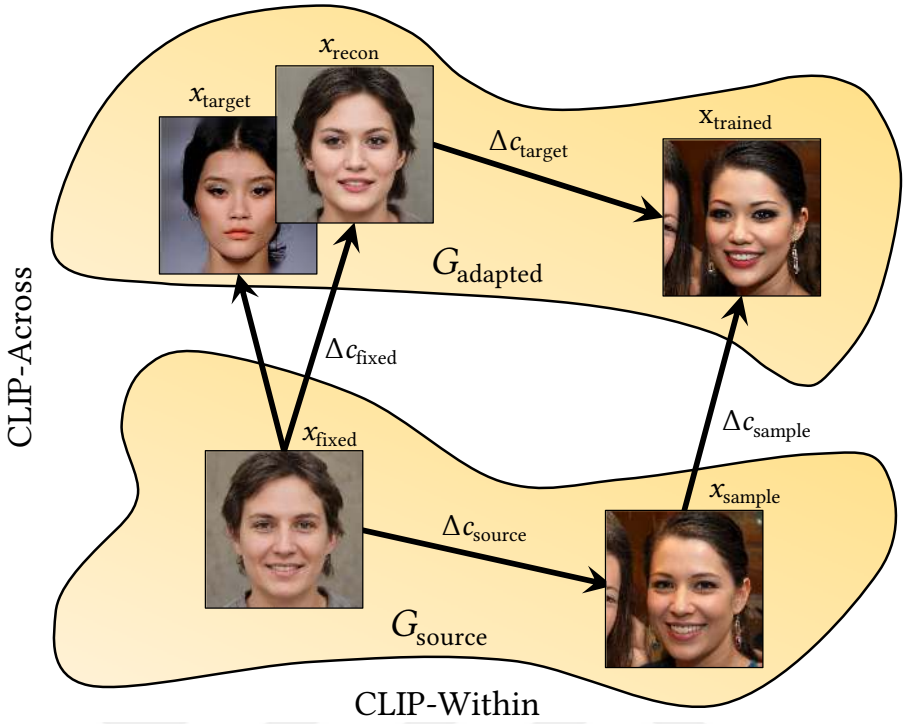

Figure 3.4: **Visualization of the directional CLIP losses for reference-guided image synthesis.** In reference-guided image synthesis, source and target domains are the same, and thus it involves in-domain adaptation. The CLIP-Across loss uses the mean StyleGAN image as the anchor image x_{fixed}. On the other hand, the CLIP-Within loss utilizes the reconstructed image x_{recon} to better preserve facial identity and image content.

guided image synthesis can be viewed as text-free training for text-guided image manipulation. With this strategy, our HyperGAN-CLIP model can effectively handle novel textual descriptions that were not encountered during training, expanding the versatility and applicability of our approach. Specifically, at inference, we can use the target text t_{target} by computing its Δ-CLIP embedding as:

$$\Delta c_{\text{text}} = \text{CLIP}(t_{\text{target}}) - \text{CLIP}(t_{\text{source}}) \tag{3.26}$$

where t_{source} can be any text matching the original unconditioned image. In our experiments, we simply set it to the word "*face*".

Chapter 4

EXPERIMENTS

We evaluate the effectiveness of our versatile framework for image synthesis and editing framework across a range of diverse tasks. We first investigate the performance of our approach in one-shot domain adaptation (Sec. 4.1). Subsequently, we conduct a comprehensive comparison with state-of-the-art reference-guided image synthesis methods highlighting the superiority of our proposed framework (Sec.4.2). Lastly, we delve into the text-guided image manipulation capabilities of our approach (Sec. 4.3). In the following, we provide representative results and refer to the supplementary material for additional results. Furthermore, while we mainly use face datasets for our evaluation, in the supplementary material we also provide reference guided and text guided editing results for the *birds* domain to demonstrate the versatility of our approach.

Training and Implementation Details For training, we use the Adam optimizer with coefficients $\beta_1 = 0.0$ and $\beta_2 = 0.99$. We set the learning rate to 0.002 and the batch size to 4. For computing the CLIP based losses, we use ViT-B/16 and ViT-B/32 CLIP encoder models and add their results as done in MTG. For input to HyperCLIP-Adapter, we use the ViT-B/16 version of the CLIP model. The scaling parameter for the modulated features is set as $\eta = 0.1$. We choose a small value for η to prevent a large shift in the distribution of features of the pretrained generator for a stable start of training. We empirically determine the weights for the individual loss terms as $\lambda_1 = 30$, $\lambda_2 = 1.5$, $\lambda_3 = 0.5$, $\lambda_4 = 0.2$, $\lambda_5 = 1.0$, $\lambda_6 = 3.0$, $\lambda_7 = 8.0$, and $\lambda_8 = 12.0$.

In domain adaptation and reference guided image synthesis experiments, in order to find x_{fixed} in the source domain corresponding to a target image, we use the e4e inversion network. However, instead of using the inversion directly, we bring it closer

to the mean latent by applying latent truncation. This prevents the inversion to lie in an out-of-distribution region and avoids x_{fixed} and x_{target} to be too close to limit meaningful editing directions.

4.1 One-Shot Domain Adaptation

Experimental Setup We expand a StyleGAN2 model pretrained on FFHQ [Karras et al., 2019] dataset to 101 new domains introduced with the expanded version of StyleGAN-NADA [Gal et al., 2022]. We generate our training data by using the expanded StyleGAN2 model shared by the authors of Domain Expansion [Nitzan et al., 2023][1]. In particular, we sample a single image for each target domain, and use these images to train our HyperGAN-CLIP model for one-shot domain adaptation. We compare HyperGAN-CLIP against the state-of-the-art domain adaptation models, namely StyleGAN-NADA [Gal et al., 2022], RSSA [Xiao et al., 2022], and DynaGAN [Kim et al., 2022b]. We train these models also in the one-shot setting with the same training data we use for our model. It is important to emphasize that both StyleGAN-NADA and RSSA require training separate models for each target domain. On the other hand, DynaGAN and our proposed HyperGAN-CLIP models are capable of modeling multiple domains with just a single model. RSSA model compresses the latent space, and the corresponding latent codes of the images are obtained using an optimization based approach as provided by II2S [Zhu et al., 2021]. Hence, it takes ∼10 hours to train the RSSA model at 256×256 pixels resolution on a single domain. In contrast, training DynaGAN and our HyperGAN-CLIP on 101 domains takes ∼12 hours in total, showing their efficiency in modeling multiple domains.

Evaluation Metrics To quantitatively assess the quality and fidelity of the generated images, we adopt the widely used Fréchet Inception Distance (FID) score [Heusel et al., 2017]. The FID score provides a measure of the statistical distance between

[1]The NADA-expanded model used in our experiments can be accessed via `https://github.com/adobe-research/domain-expansion/tree/main`.

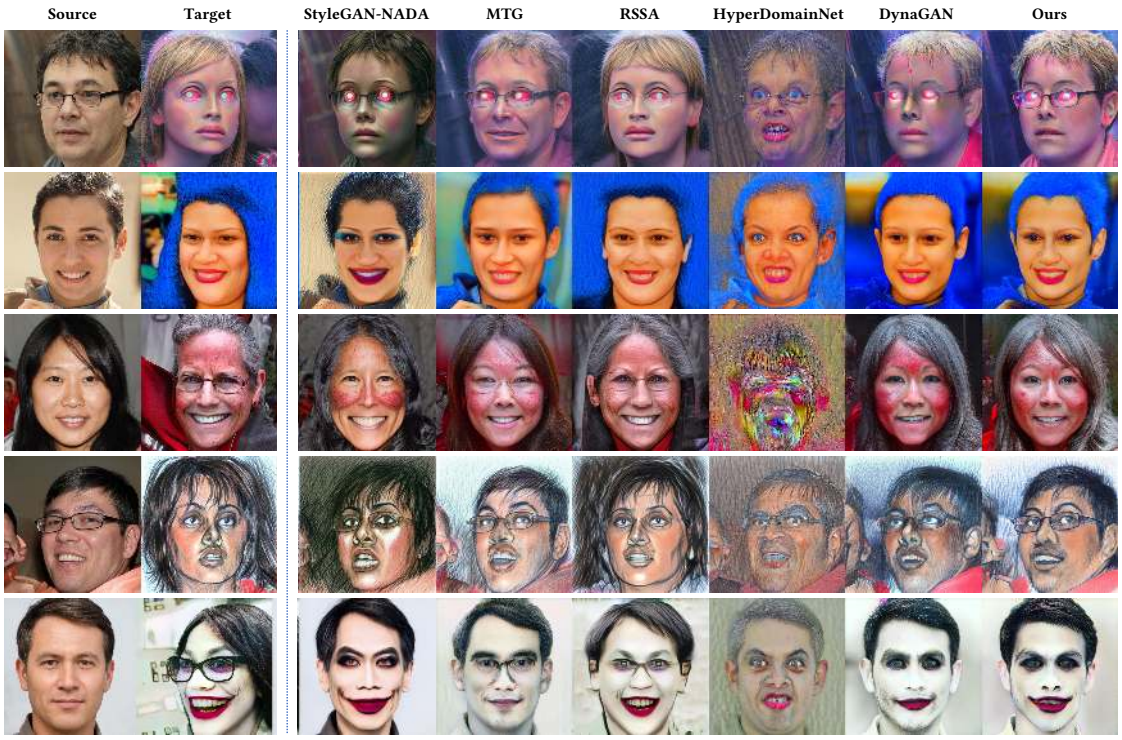

Figure 4.1: **Comparison against the state-of-the-art few-shot domain adaptation methods**. Our proposed HyperGAN-CLIP model outperforms competing methods in accurately capturing the visual characteristics of the target domains. The synthesized images exhibit a higher degree of fidelity and realism, demonstrating the effectiveness of our approach.

the distributions of real and generated images. In our evaluation, we generate a set of 1K images for each target domain using the NADA-expanded Domain Expansion model, treating these images as real. For the FID evaluation, we compare the distribution of these images with that of images generated by the evaluated methods. Specifically, we randomly sample 100 images from each target domain to represent the generated image distribution.

Results In Fig. 4.1, we show sample images generated by the models adapted using the evaluated domain-adaptation techniques. For each sample, we present the source image alongside the corresponding training image representing the target domain, and the synthesized images. Notably, StyleGAN-NADA falls short in fully capturing the visual characteristics of the target domains, giving low-fidelity

results. MindTheGAP works fairly well with better quality, but it still tends to overfit slightly to the target domain image (e.g. eyes, glasses etc.). While RSSA is capable of modeling structures from the target domain, it suffers from severe mode collapse. The generated images closely resemble the target image in terms of identity, differing only in pose. HyperDomainNet generates a lot of artifacts after adaptation even though it has been shown to work well for text prompts instead of target images. DynaGAN demonstrates improved performance compared to these methods, but it still exhibits some unnatural artifacts. However, when combined with the additional style mixing strategy, these artifacts are effectively mitigated. It is important to note that DynaGAN struggles to retain the identity of the source image effectively. Our proposed HyperGAN-CLIP model surpasses the visual quality of the other approaches, thanks to our novel adaptation module that leverages CLIP-guided hypernetworks. Our approach also takes into account the shared characersitics between different domains and also prevents overfitting. The resulting images exhibit remarkable visual fidelity as also shown by the superiority of our FID metrics shown in Table 4.1[2].

Ablation study We conduct an ablation study to assess the impact of each component in our model. The qualitative and quantitative results are presented in Fig. 4.2 and Table 4.2, respectively. The baseline network uses only the features given by the target-domain modulated generator, and ignores the source domain features. This approach results in the loss of the source identity and is prone to overfitting to the provided target image. Adding a conditional discriminator loss helps to mitigate the problems to some extent and enhances image quality. Considering residual features scheme that employs target features alongside with the source features preserves the facial identity better than the baseline, but falls short in terms of image quality. Finally, our full model, HyperGAN-CLIP, which utilizes residual feature scheme together with a conditional discriminator effectively preserves identity while

[2]We compute the FID score of RSSA on a subset of 24 domains, considering images at a resolution of 256×256 pixels, as a separate RSSA model needs to be trained for each domain and the training is too costly.

Table 4.1: **Quantitative comparisons for one-shot domain adaptation.** Our HyperGAN-CLIP model produces images closely resembling the target domain images, as evidenced by the second-lowest FID score, showing the superior performance of our approach in capturing the target domain characteristics. The best-performing and the second best-performing models are indicated in bold and underlined typeface, respectively. The best-performing in terms of FID is visually not the best, however.

Method	FID ↓
StyleGAN-NADA [Gal et al., 2022]	22.80
MindTheGap [Zhu et al., 2022]	**16.04**
DynaGAN [Kim et al., 2022b]	21.46
RSSA [Xiao et al., 2022]	67.01
HyperDomainNet [Alanov et al., 2023]	59.73
Ours	<u>16.46</u>

capturing target style and maintaining high image quality.

Table 4.2: **Quantitative analysis of the ablation study.** The baseline model that employs target-domain modulated features gives the worst score. However, incorporating CLIP-conditioned discriminator and leveraging residual features scheme introduce notable improvements. Our full HyperCLIP-GAN model, utilizing all these components achieves the best score.

Component	FID ↓
Baseline (B)	23.40
Baseline + Cond. Disc. (B + CD)	18.19
Baseline + Res. Features (B + RF)	18.01
HyperGAN-CLIP (B + CD + RF)	**16.46**

Controllable Manipulation We observe that scaling the CLIP embeddings translates roughly to scaling of the modulation weights, and consequently, feature maps. By adjusting the scaling ratio of the residual target domain features injected into the

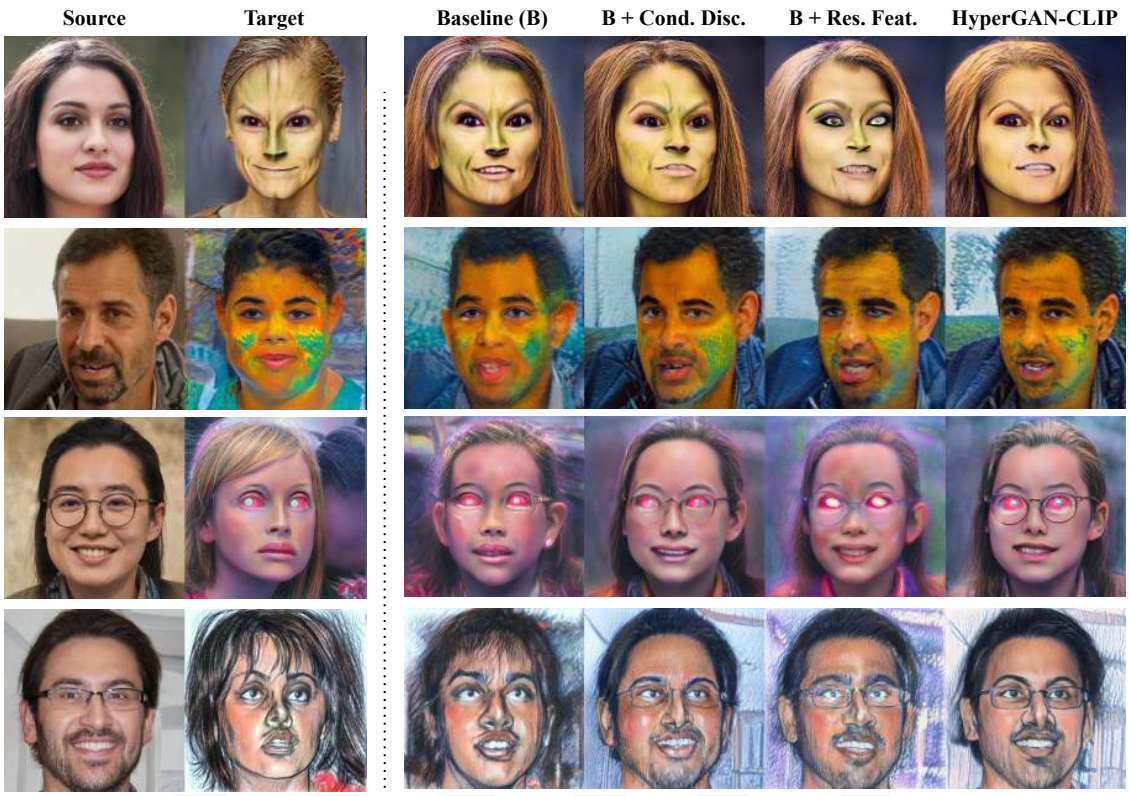

Figure 4.2: **Qualitative results for the ablation study.** Baseline network does not preserve the facial identity of the source image, giving an outcome closely resembling to the target image. When CLIP-conditioned discriminator is incorporated to the baseline, the image quality is improved. Using residual features scheme preserves the facial identity better. Our full model gives the best results in terms of both identity and image quality.

source domain features by a factor β, we can control the degree of adaptation during inference. Furthermore, in line with previous GAN domain adaptation studies, we can enhance the style quality by employing style mixing over the latent codes. This involves interpolating the original latent code w with the target domain's latent code w_t as $\hat{w} = \alpha * w + (1 - \alpha) * w_t$, where α is a scalar between 0 and 1, controlling the level of style mixing. The resulting interpolated latent code $\hat{w}$ can be then used as input to the generator for image synthesis. We demonstrate the impact of these control parameters on the generated images in Fig. 4.3.

Moreover, we observe that even if we additionally scale the amount of features injected into the original image features by some factor β, our approach gives very

Figure 4.3: **Controllable Manipulation.** In our approach, we can vary the amount of residual features injected as well as the amount of target style latent, which gives users the ability to control degree of adaptation with respect to style consistency vs data fidelity.

plausible results. It does not change the features that not related to the target. We compare it with our baseline model with conditional discriminator trained with Δ-CLIP embeddings, where we use the same β to scale the modulation parameters to control the degree of adaptation (as in DynaGAN). Since, it does not use original domain features and weight modulation is responsible for preserving both original image characteristics and transferring target style content from CLIP embeddings,

it fails to scale well with amount of the feature scaling parameter, as demonstrated in Fig. 4.4. This highlights the importance of using domain specific features as residuals to the original features instead of directly generating the overall combined features.

Domain Mixing By leveraging the structured CLIP embedding space as a conditioning mechanism, our approach enables smooth interpolation between different domain embeddings. This capability allows us to generate images that combine elements from multiple domains and even explore novel compositions of domain embeddings within the distribution of the training set. In Fig. 4.5, we give an example to this domain mixing strategy.

As also demonstrated in our text-based image manipulation experiments in Section 4.3, this generalization ability to novel compositions improves as more target domains are included in the training set. Notably, our ability to generate and edit images using novel captions highlights the capacity of our approach to generalize to previously unseen domains.

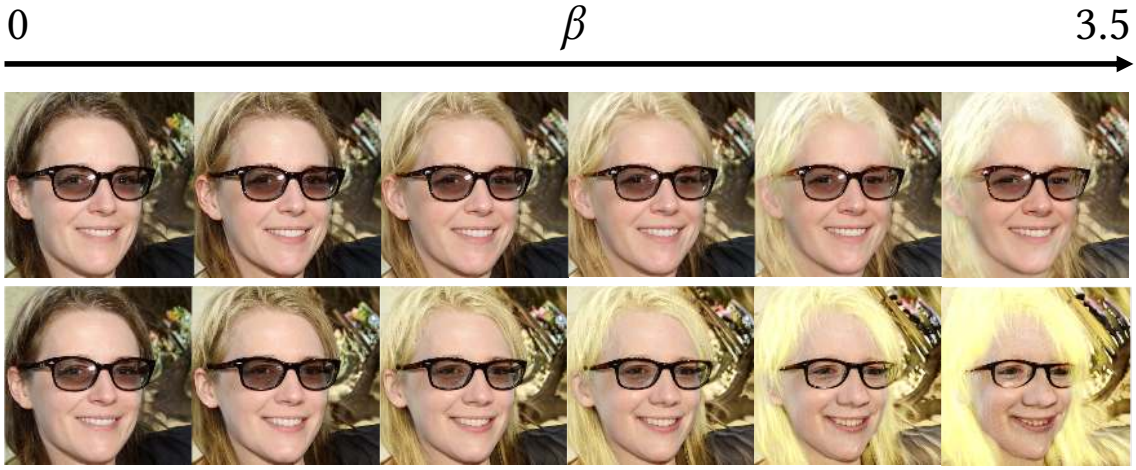

Figure 4.4: **Scaling residual features.** The top row shows the results obtained with our approach, whereas the bottom on corresponds to the results by the baseline model with the discriminator. Several artifacts instantly start to appear in the baseline results when scaling beyond the training value of 1 (third column corresponds to $\beta = 1$). On the other hand, our approach works relatively seamlessly and preserves the identity better.

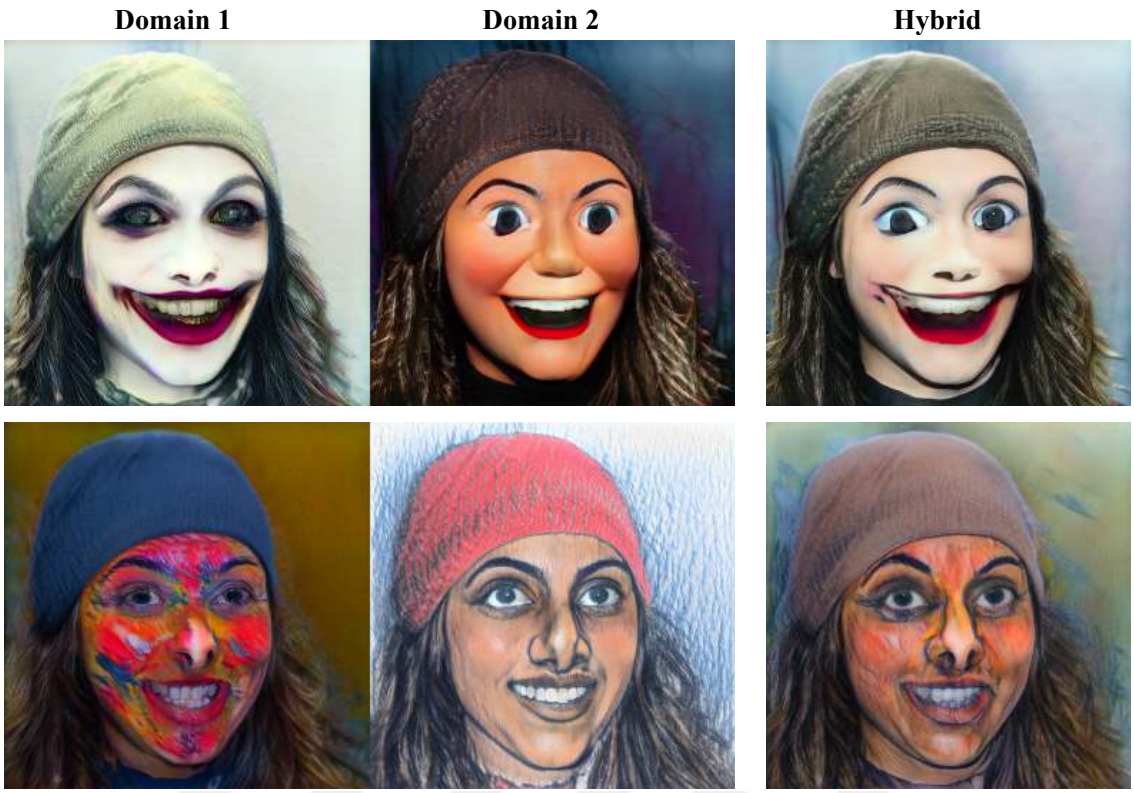

Figure 4.5: **Domain mixing.** HyperCLIP-Adapter module enables the fusion of multiple domains, allowing for the generation of novel compositions. By combining CLIP embeddings of two target domains by taking their mean and re-scaling, we generate images that exhibit characteristics from both domains.

Editing in Target Domains Our framework facilitates a shared latent space between the source and the adapted target domains, enabling the utilization of meaningful latent directions discovered in the GAN space for semantic editing of target domain images. Fig.4.6 provides an illustrative example of this kind of semantic editing, showing the ability to modify attributes such as age, smile, and pose using InterfaceGAN [Shen et al., 2020b]. This demonstrates the practicality of leveraging existing techniques to perform effective semantic edits within our framework.

4.2 Reference-Guided Image Synthesis

Experimental Setup In this experiment, our objective is to synthesize a new image that combines the identity of a source image with the style of a target image,

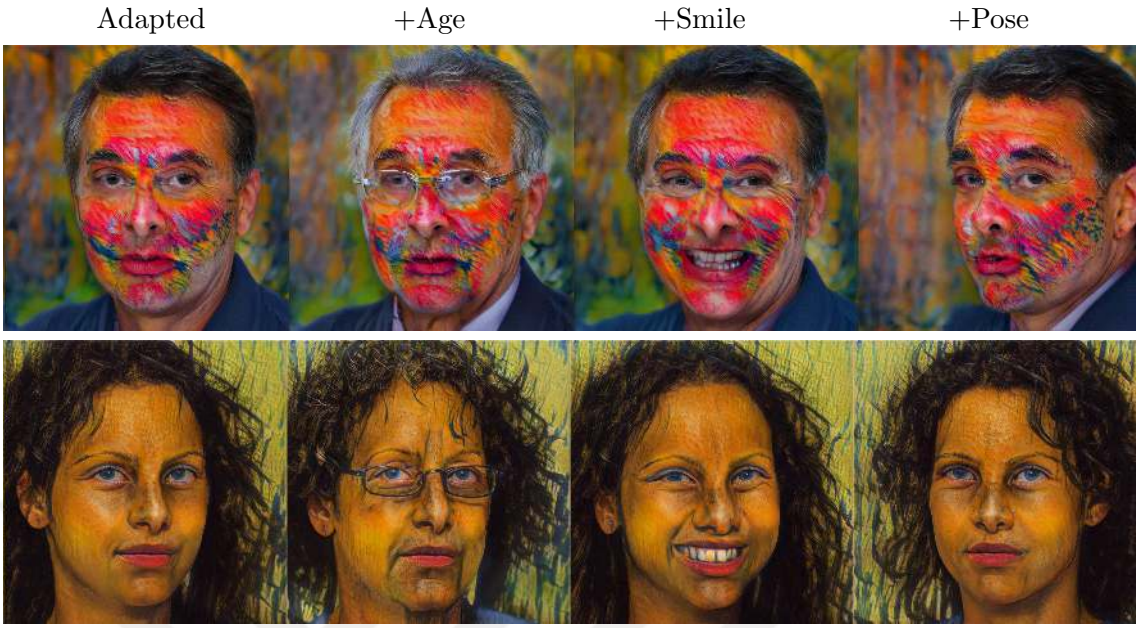

Figure 4.6: **Semantic editing in target domains.** Since latent mapper is kept intact, our approach allows for using existing latent space discovery methods to perform semantic edits. We manipulate two sample face images from adapted domains by playing with age, smile, and pose.

as represented by its CLIP embedding. To achieve this, we sample the source images from the StyleGAN2 generator pre-trained on the FFHQ dataset, and utilize the test set of the CelebA-HQ dataset [Lee et al., 2020], which comprises a total of 6000 diverse images, as our pool of target images. We compare our HyperGAN-CLIP model against three state-of-the-art approaches in qualitative and quantitative manner. These approaches are BlendGAN [Liu et al., 2021a], TargetCLIP-O [Chefer et al., 2022] and TargetCLIP-E [Chefer et al., 2022]. While BlendGAN and TargetCLIP-E are encoder-based approaches, TargetCLIP-O employs a direct optimization scheme for reference-guided image synthesis. Our approach, apart from these studies, is based on modulating the StyleGAN generator via CLIP-guided hypernetworks.

Evaluation Metrics We use FID [Heusel et al., 2017] to measure the quality and the fidelity of the synthesized images as a lower FID score indicates that the synthesized images are closer to the FFHQ domain, which is the original domain the StyleGAN2 is trained on. We use the ID similarity [Deng et al., 2022] to measure

Figure 4.7: **Qualitative comparison with state-of-the-art reference-guided image synthesis approaches.** Our approach effectively transfers the style of the target image to the source image while effectively preserving identity compared to competing methods.

identity preservation. Ideally, we want the ID similarity with the source image to be high and ID similarity with the target image to be low as we want to preserve the identity of the source image while only transferring the attributes of the target image to the source. Finally, we use the CLIP embedding space to measure the semantic similarity of the target and output images to evaluate how well the semantics of the target image are transferred.

Results In Fig. 4.7, we provide qualitative comparisons with prior reference-guided image synthesis methods. Sample source-target pairs show a diverse range of visual characteristics in terms of gender, age, hair color, ethnicity. We observe that BlendGAN tends to produce cartoon-like outputs that lack naturalness. the optimization-based variant of TargetCLIP (TargetCLIP-O) demonstrates superior

performance compared to its encoder-based counterpart (TargetCLIP-E) in maintaining identity while incorporating the desired style changes depicted in the target image. Notably, our HyperGAN-CLIP approach excels in seamlessly transferring the attributes from the chosen target faces to the source faces while preserving identity to a greater extent than the competing methods. These results affirm the effectiveness of our approach in generating visually compelling outputs with enhanced fidelity and plausibility.

Table 4.3 presents the quantitative scores comparing our approach with other existing methods. Notably, our method achieves competitive results in terms of FID, comparable to the performance of TargetCLIP-O, which performs latent optimization for each target. This highlights the ability of our method to synthesize high-quality and faithful images. Furthermore, our approach outperforms competing methods in preserving the identity of the source image, as indicated by the ID similarity scores. Additionally, our method excels in CLIP semantic similarity, affirming its capability to capture the semantics of the target image in the synthesized results. Overall, our approach strikes a favorable balance across multiple evaluation metrics, showing its effectiveness in photo-realistic image synthesis and maintaining key visual attributes.

4.3 Text-Guided Image Manipulation

Experimental Setup In this experiment, we show the versatility of our proposed framework by demonstrating its ability to manipulate input images based on target textual descriptions. To conduct this experiment, we leverage the CelebA dataset's test set [Liu et al., 2015] along with its attribute annotations. We select attributes that are absent from the images and construct target descriptions that prompt the desired attribute manipulation. Leveraging a pre-trained e4e model [Tov et al., 2021], we perform an image-to-latent-space inversion, generating latent representations of the input images. These inverted images serve as inputs to our framework. To condition the synthesis process, we utilize Δ-CLIP embeddings, which capture the discrepancy between the CLIP embeddings of the target description and

Table 4.3: **Quantitative comparisons for reference-guided image synthesis on the CelebA-HQ dataset.** Our multi-purpose framework achieves comparable results against the state-of-the-art in reference-guided image synthesis. Our method synthesizes images with high fidelity, as indicated by the low FID scores, while effectively preserving the identity of the source image. Furthermore, our method successfully transfers the semantic details of the target image to the source, as proven by the CLIP similarity score. For each metric, the best-performing and the second best-performing models are indicated in bold and underlined typeface, respectively.

Method	FID ↓	ID (source) ↑	ID (target) ↓	CLIP Sim. ↑
BlendGAN	69.92	<u>36.13±10.40</u>	**2.45±9.61**	70.07±9.33
TargetCLIP-O	**24.49**	31.60±20.27	15.91±10.58	<u>72.76±11.65</u>
TargetCLIP-E	41.76	22.19±13.30	29.36±10.61	68.38±9.33
Ours	<u>26.73</u>	**72.45±7.63**	<u>10.93±9.73</u>	**90.61±4.82**

the input image. We perform a comprehensive comparison of our method against several state-of-the-art text-guided image manipulation approaches. These include TediGAN-B [Xia et al., 2021], StyleCLIP-LO [Patashnik et al., 2021], StyleCLIP-GD [Patashnik et al., 2021], HairCLIP [Wei et al., 2022], DeltaEdit [Lyu et al., 2023], and CLIPInverter [Hidden, 2023] as representative GAN-based methods. Among these, DeltaEdit is the only model that utilizes text-free training like our method. Additionally, we also compare against diffusion-based approaches, namely Diffusion-CLIP [Kim et al., 2022a] and Plug-and-Play [Tumanyan et al., 2022]. Among these, the method most similar to ours is DeltaEdit in the sense that it is also solely on image data and does not utilize any text data during training. By evaluating our method against these diverse approaches, we provide a comprehensive analysis of its performance and highlight its distinct advantages in text-guided image manipulation.

Evaluation Metrics To evaluate the approaches quantitatively, we employ multiple quantitative metrics, namely Fréchet Inception Distance (FID) [Heusel et al.,

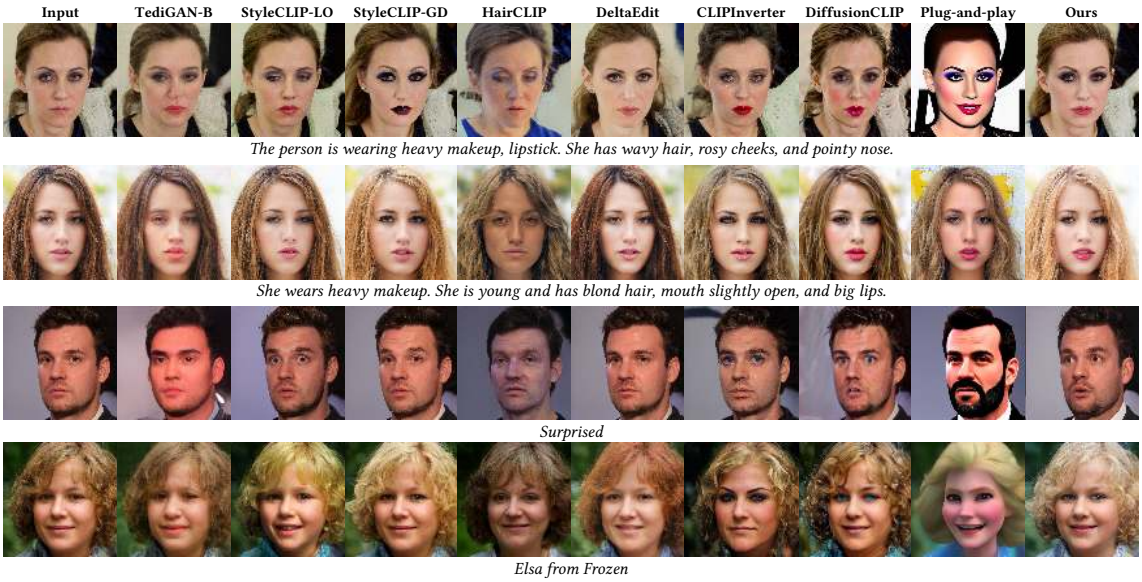

Figure 4.8: **Qualitative comparisons with state-of-the-art text-guided image manipulation methods.** Our model shows remarkable versality in manipulating images across a diverse range of textual descriptions. The results vividly illustrate our model's ability to accurately apply changes based on target descriptions encompassing both single and multiple attributes. Compared to the competing approaches, our model preserves the identity of the input much better while successfully executing the desired manipulations.

2017], Attribute Manipulation Accuracy (AMA), and CLIP Manipulative Precision (CMP) following the methodology introduced by CLIPInverter [Hidden, 2023]. FID serves as a measure of the quality and fidelity of the synthesized images. To assess AMA, we select 15 attribute annotations from the CelebA dataset [Liu et al., 2015] and manipulate 50 images for each attribute. Subsequently, the manipulated images are passed through attribute classifiers to determine the average manipulation accuracy. We refer to the supplementary material for more details on the AMA computation. In order to quantify the alignment between the output images and the target captions, while preserving the contents of the input image, we employ CMP, which is defined as $\text{CMP} = (1 - \text{diff}) \cdot \text{sim}$, with diff denoting the L1 pixel difference between the input and output images, and sim denotes the CLIP semantic similarity between the output image and the target description.

Results Figure 4.8 presents text-guided image manipulation results of our proposed approach along with several competing methods across various textual descriptions. TediGAN-B and DeltaEdit struggle to effectively manipulate the images, often resulting in images similar to the input. While StyleCLIP-LO and HairCLIP perform better, they still exhibit limitations by manipulating unwanted attributes in certain cases (e.g., rows 2 and 4). StyleCLIP-GD encounters difficulties in challenging scenarios, as illustrated by the description *surprised* given in the third row. CLIPInverter performs well when explicit attribute manipulations are specified in the descriptions (first two rows), but it falls short when encountering novel descriptions unseen during its training, such as surprised. DiffusionCLIP generates images with noticeable artifacts, leading to poor output quality. While Plug-and-play successfully applies most manipulations, the resulting images often lack realism, appearing cartoonish and with unintended attribute modifications. In contrast, our model, even trained without any textual data, successfully applies single or multiple attribute changes while better preserving the identity of the input images compared to the competing approaches.

Table 4.4 presents insightful quantitative comparisons between our approach and the competing methods. Here, we group our approach and DeltaEdit together to distinguish these works from the others which utilize additional text data during training. We evaluate manipulation accuracy and precision using AMA (Single) for single attribute changes and AMA (Multiple) for multiple attribute changes. Remarkably, our model achieves comparable or even better performance in manipulation accuracy and precision compared to leading text-guided image manipulation models, including StyleCLIP, and DiffusionCLIP. In terms of FID, the diffusion-based models, DiffusionCLIP and Plug-and-play, excel as compared to GAN-based approaches due to their high-quality generation capabilities. Even though we do not use textual data during training, our model finds a good balance between the metrics and consistently delivers competitive performance. It effectively handles descriptions involving multiple attribute changes. More importantly, as compared to DeltaEdit, the other text-guided image manipulation method with text-free training, our HyperGAN-CLIP gives much superior performance.

Table 4.4: **Quantitative comparisons for text-guided image manipulation.** Despite not being explicitly trained on textual descriptions, our HyperGAN-CLIP consistently achieves highly competitive results compared to state-of-the-art text-guided image editing methods. For each metric, the best-performing and the second best-performing models are indicated in bold and underlined typeface, respectively.

	FID ↓	CMP ↑	AMA ↑ (Single)	AMA ↑ (Multiple)
TediGAN-B	<u>55.424</u>	**0.285**	11.286	1.142
StyleCLIP-LO	80.833	0.210	15.857	3.429
StyleCLIP-GD	82.393	0.191	33.143	11.429
HairCLIP	93.523	0.218	<u>41.571</u>	<u>15.149</u>
CLIPInverter	97.210	0.221	**61.429**	**41.714**
DiffusionCLIP	**29.280**	<u>0.243</u>	26.000	4.857
Plug-and-play	68.287	0.199	27.429	7.143
DeltaEdit	80.316	0.171	8.857	0.571
Ours	87.851	0.189	25.143	10.000

Impact of Δ-CLIP Embeddings As previously mentioned, the Δ-CLIP space provides a semantic embedding space that offers improved alignment between text and image modalities compared to the original CLIP space. This distinction becomes particularly evident when examining the text-guided image manipulation task. In Fig. 4.9, we show the impact of utilizing these spaces within our HyperCLIP-Adapter module. The results are striking, as the Δ-CLIP embeddings enable highly precise text-based editing while preserving image quality and identity fidelity. We provide additional results in the supplementary material.

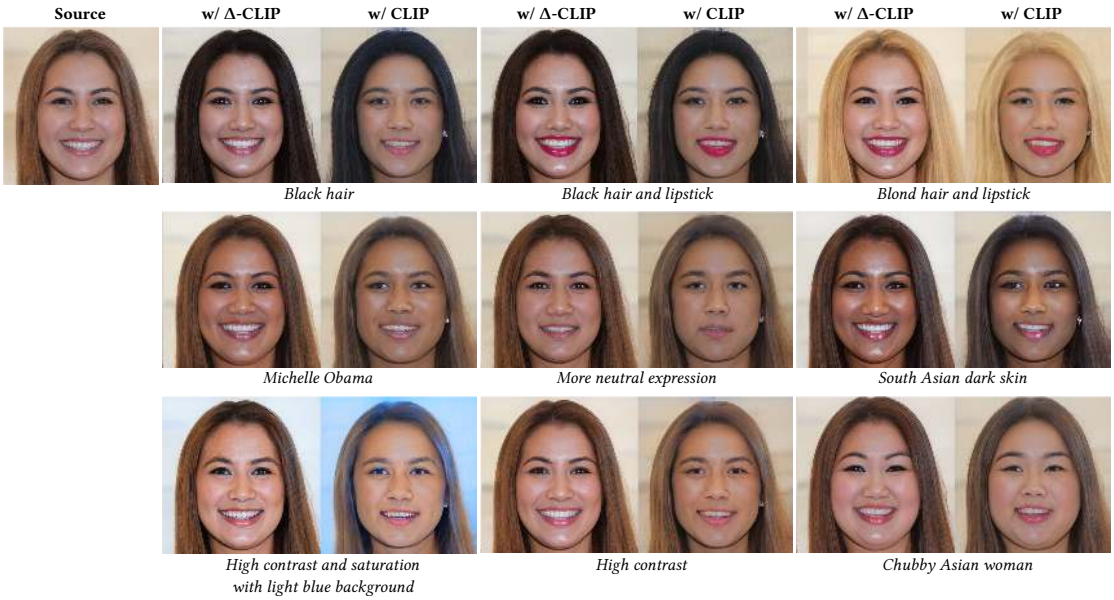

Figure 4.9: **Impact of Δ-CLIP Embeddings.** Our model equipped with Δ-CLIP embeddings performs semantic edits that are better aligned with the provided textual descriptions as compared to the version of our model that employs original CLIP embeddings.

Chapter 5

CONCLUSION

In this paper, we present a novel and flexible framework for one-shot domain adaptation of GANs capable of handling multiple target domains. Our approach leverages a single lightweight hypernetwork to dynamically adapt a StyleGAN generator using both images and text. By employing residual feature injection and conditional discriminator, we ensure preserving image diversity and source identity while effectively transferring the characteristics of the target domain. Using conditional discriminator also contributes to the generation of high-fidelity images. Through extensive quantitative and qualitative evaluation, we demonstrate the superior performance of our approach compared to competing methods in the domain adaptation task. When it comes to reference-guided image synthesis, our approach achieves the best results while maintaining the essence of the source identity. Furthermore, in text-based image editing, our approach outperforms other text-free training methods while remaining competitive with approaches trained on text data.

Limitations and future work In our current approach, we employ separate models for domain adaptation and text/image-based editing. However, an interesting research direction could be developing a unified model that can handle both tasks seamlessly. By leveraging pre-training on image/text-based editing tasks, we can transfer valuable knowledge to enhance the domain adaptation process. During training, we can incorporate regularization techniques to ensure control over both in-domain and out-of-domain target images. Moreover, including cyclic losses or residual feature norm regularization holds promise for further improving the performance and stability of the model. Additionally, there is a growing interest in exploring dual-space GAN models that offer improved disentanglement of style and content information [Kwon and Ye, 2021, Xu et al., 2022], compared to the original

StyleGAN model. It could be interesting to explore incorporating our framework to these models, as it might have the potential to enhance the effectiveness of domain adaptation results.

BIBLIOGRAPHY

[Abdal et al., 2019] Abdal, R., Qin, Y., and Wonka, P. (2019). Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4431–4440.

[Abdal et al., 2020] Abdal, R., Qin, Y., and Wonka, P. (2020). Image2stylegan++: How to edit the embedded images? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[Abdal et al., 2021] Abdal, R., Zhu, P., Mitra, N. J., and Wonka, P. (2021). Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Trans. Graph.*, 40(3).

[Alaluf et al., 2021] Alaluf, Y., Patashnik, O., and Cohen-Or, D. (2021). Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[Alaluf et al., 2022] Alaluf, Y., Tov, O., Mokady, R., Gal, R., and Bermano, A. (2022). Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18511–18521.

[Alanov et al., 2023] Alanov, A., Titov, V., and Vetrov, D. (2023). Hyperdomainnet: Universal domain adaptation for generative adversarial networks.

[Bai et al., 2022] Bai, Q., Xu, Y., Zhu, J., Xia, W., Yang, Y., and Shen, Y. (2022). High-fidelity gan inversion with padding space. In *European Conference on Computer Vision*, pages 36–53. Springer.

[Bau et al., 2019a] Bau, D., Strobelt, H., Peebles, W., Wulff, J., Zhou, B., Zhu, J.-Y., and Torralba, A. (2019a). Semantic photo manipulation with a generative image prior. *ACM Trans. Graph.*, 38(4).

[Bau et al., 2019b] Bau, D., Zhu, J.-Y. Z., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., and Torralba, A. (2019b). Inverting layers of a large generator. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[Brock et al., 2019] Brock, A., Donahue, J., and Simonyan, K. (2019). Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

[Chefer et al., 2022] Chefer, H., Benaim, S., Paiss, R., and Wolf, L. (2022). Image-based clip-guided essence transfer. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, page 695–711.

[Chen et al., 2021] Chen, S.-Y., Liu, F.-L., Lai, Y.-K., Rosin, P. L., Li, C., Fu, H., and Gao, L. (2021). DeepFaceEditing: Deep generation of face images from sketches. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2021)*, 40(4):90:1–90:15.

[Chen et al., 2020] Chen, X., Fan, H., Girshick, R., and He, K. (2020). Improved baselines with momentum contrastive learning.

[Creswell and Bharath, 2019] Creswell, A. and Bharath, A. A. (2019). Inverting the generator of a generative adversarial network. *IEEE Transactions on Neural Networks and Learning Systems*, 30(7):1967–1974.

[Deng et al., 2022] Deng, J., Guo, J., Yang, J., Xue, N., Kotsia, I., and Zafeiriou, S. (2022). ArcFace: Additive angular margin loss for deep face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):5962–5979.

[Dinh et al., 2022] Dinh, T. M., Tran, A. T., Nguyen, R., and Hua, B.-S. (2022). Hyperinverter: Improving stylegan inversion via hypernetwork. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Gal et al., 2022] Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Trans. Graph.*, 41(4).

[Gatys et al., 2015] Gatys, L. A., Ecker, A. S., and Bethge, M. (2015). A neural algorithm of artistic style. *ArXiv*, abs/1508.06576.

[Goodfellow et al., 2014a] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014a). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

[Goodfellow et al., 2014b] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014b). Generative adversarial networks.

[Ha et al., 2017] Ha, D., Dai, A. M., and Le, Q. V. (2017). Hypernetworks. In *International Conference on Learning Representations*.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition.

[Heusel et al., 2017] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

[Hidden, 2023] Hidden, A. (2023). Clip-guided stylegan inversion for text-driven real image editing. *Conditionally accepted to ACM TOG*.

[Hou and Kung, 2021] Hou, Z. and Kung, S.-Y. (2021). Parameter efficient dynamic convolution via tensor decomposition. In *Proceedings of the British Machine Vision Conference (BMVC)*.

[Huang and Belongie, 2017] Huang, X. and Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization.

[Hudson and Zitnick, 2021] Hudson, D. A. and Zitnick, C. L. (2021). Generative adversarial transformers. *Proceedings of the 38th International Conference on Machine Learning, ICML 2021*.

[Härkönen et al., 2020] Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. (2020). Ganspace: Discovering interpretable gan controls. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

[Jiang et al., 2022] Jiang, K., Chen, S.-Y., Liu, F.-L., Fu, H., and Gao, L. (2022). Nerffaceediting: Disentangled face editing in neural radiance fields. In *ACM SIGGRAPH Asia 2022 Conference Proceedings*, SIGGRAPH Asia'22, New York, NY, USA. Association for Computing Machinery.

[Karras et al., 2018] Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*.

[Karras et al., 2021] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., and Aila, T. (2021). Alias-free generative adversarial networks. In *Proc. NeurIPS*.

[Karras et al., 2019] Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Karras et al., 2020] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Kim et al., 2022a] Kim, G., Kwon, T., and Ye, J. C. (2022a). Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2426–2435.

[Kim et al., 2022b] Kim, S., Kang, K., Kim, G., Baek, S.-H., and Cho, S. (2022b). Dynagan: Dynamic few-shot adaptation of gans to multiple domains. In *Proceedings of the ACM (SIGGRAPH Asia)*.

[Kocasari et al., 2021] Kocasari, U., Dirik, A., Tiftikci, M., and Yanardag, P. (2021). Stylemc: Multi-channel based fast text-guided image generation and manipulation. In *WACV*.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Proc. NeurIPS*, pages 1097–1105.

[Kumari et al., 2022] Kumari, N., Zhang, R., Shechtman, E., and Zhu, J.-Y. (2022). Ensembling off-the-shelf models for gan training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Kwon and Ye, 2021] Kwon, G. and Ye, J. C. (2021). Diagonal attention and style-based GAN for content-style disentanglement in image generation and translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[Lee et al., 2020] Lee, C.-H., Liu, Z., Wu, L., and Luo, P. (2020). Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Littwin and Wolf, 2019] Littwin, G. and Wolf, L. (2019). Deep meta functionals for shape representation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[Liu et al., 2021a] Liu, M., Li, Q., Qin, Z., Zhang, G., Wan, P., and Zheng, W. (2021a). Blendgan: Implicitly gan blending for arbitrary stylized face generation. In *Advances in Neural Information Processing Systems*.

[Liu et al., 2021b] Liu, X., Gong, C., Wu, L., Zhang, S., Su, H., and Liu, Q. (2021b). Fusedream: Training-free text-to-image generation with improved clip+gan space optimization. *arXiv preprint arXiv:2112.01573*.

[Liu et al., 2015] Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

[Lyu et al., 2023] Lyu, Y., Lin, T., Li, F., He, D., Dong, J., and Tan, T. (2023). Deltaedit: Exploring text-free training for text-driven image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6894–6903.

[Miyato and Koyama, 2018] Miyato, T. and Koyama, M. (2018). cGANs with projection discriminator. In *International Conference on Learning Representations*.

[Nitzan et al., 2023] Nitzan, Y., Gharbi, M., Zhang, R., Park, T., Zhu, J.-Y., Cohen-Or, D., and Shechtman, E. (2023). Domain expansion of image generators.

[Ojha et al., 2021] Ojha, U., Li, Y., Lu, C., Efros, A. A., Lee, Y. J., Shechtman, E., and Zhang, R. (2021). Few-shot image generation via cross-domain correspondence. In *CVPR*.

[Patashnik et al., 2021] Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., and Lischinski, D. (2021). Styleclip: Text-driven manipulation of stylegan imagery.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2085–2094.

[Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

[Richardson et al., 2021] Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., and Cohen-Or, D. (2021). Encoding in style: a stylegan encoder for image-to-image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Shen et al., 2020a] Shen, Y., Gu, J., Tang, X., and Zhou, B. (2020a). Interpreting the latent space of gans for semantic face editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Shen et al., 2020b] Shen, Y., Yang, C., Tang, X., and Zhou, B. (2020b). Interfacegan: Interpreting the disentangled face representation learned by gans. *TPAMI*.

[Shen and Zhou, 2021] Shen, Y. and Zhou, B. (2021). Closed-form factorization of latent semantics in gans. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Spurek et al., 2021] Spurek, P., Kasymov, A., Mazur, M., Janik, D., Tadeja, S., Łukasz Struski, Tabor, J., and Trzciński, T. (2021). Hyperpocket: Generative point cloud completion.

[Tewari et al., 2020] Tewari, A., Elgharib, M., R, M. B., Bernard, F., Seidel, H.-P., Pérez, P., Zollhöfer, M., and Theobalt, C. (2020). Pie: Portrait image embedding for semantic control. *ACM Trans. Graph.*, 39(6).

[Tov et al., 2021] Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., and Cohen-Or, D. (2021). Designing an encoder for stylegan image manipulation. *ACM Trans. Graph.*, 40(4).

[Tumanyan et al., 2022] Tumanyan, N., Geyer, M., Bagon, S., and Dekel, T. (2022). Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*.

[von Oswald et al., 2020] von Oswald, J., Henning, C., Grewe, B. F., and Sacramento, J. (2020). Continual learning with hypernetworks. In *International Conference on Learning Representations*.

[Voynov and Babenko, 2020] Voynov, A. and Babenko, A. (2020). Unsupervised discovery of interpretable directions in the gan latent space. In *International Conference on Machine Learning*, pages 9786–9796. PMLR.

[Wah et al., 2011] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

[Wei et al., 2022] Wei, T., Chen, D., Zhou, W., Liao, J., Tan, Z., Yuan, L., Zhang, W., and Yu, N. (2022). Hairclip: Design your hair by text and reference image. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[Wu et al., 2021] Wu, Z., Lischinski, D., and Shechtman, E. (2021). StyleSpace Analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12863–12872.

[Xia et al., 2021] Xia, W., Yang, Y., Xue, J.-H., and Wu, B. (2021). Tedigan: Text-guided diverse face image generation and manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Xiao et al., 2022] Xiao, J., Li, L., Wang, C., Zha, Z.-J., and Huang, Q. (2022). Few shot generative model adaption via relaxed spatial structural alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11204–11213.

[Xu et al., 2022] Xu, Y., Yin, Y., Jiang, L., Wu, Q., Zheng, C., Loy, C. C., Dai, B., and Wu, W. (2022). TransEditor: Transformer-based dual-space GAN for highly controllable facial editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Yoo et al., 2019] Yoo, J., Uh, Y., Chun, S., Kang, B., and Ha, J.-W. (2019). Photorealistic style transfer via wavelet transforms. In *International Conference on Computer Vision (ICCV)*.

[Yuval Nirkin, 2021] Yuval Nirkin, Lior Wolf, T. H. (2021). Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation. In *Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[Zhang et al., 2022] Zhang, B., Gu, S., Zhang, B., Bao, J., Chen, D., Wen, F., Wang, Y., and Guo, B. (2022). Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11304–11314.

[Zhao et al., 2020] Zhao, S., Liu, Z., Lin, J., Zhu, J.-Y., and Han, S. (2020). Differentiable augmentation for data-efficient gan training.

[Zhu et al., 2020] Zhu, J., Shen, Y., Zhao, D., and Zhou, B. (2020). In-domain gan inversion for real image editing. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[Zhu et al., 2016] Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[Zhu et al., 2022] Zhu, P., Abdal, R., Femiani, J., and Wonka, P. (2022). Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks. In *International Conference on Learning Representations*.

[Zhu et al., 2021] Zhu, P., Abdal, R., Qin, Y., Femiani, J., and Wonka, P. (2021). Improved stylegan embedding: Where are the good latents?

Appendix A

The purpose of this document is to provide extra material to complement the main paper. Section A.1 gives the details about the AMA metric calculation for our experiments. Section A.2 presents additional visual comparisons between our proposed model against the existing approaches in few-shot domain adaptation, reference-guided image synthesis and text-guided image manipulation. Section A.3 includes reference-guided editing and text-guided editing results using our HyperGAN-CLIP approach for domains such as cat and bird images. Finally, Section A.4 demonstrates additional qualitative comparisons between using raw CLIP embeddings and Δ-CLIP embeddings for text-guided editing.

A.1 AMA Score Details

Attribute Manipulation Accuracy (AMA) [Hidden, 2023] measures how well a single manipulation is applied. To calculate the AMA score of a model, for each attribute (such as *blonde hair*), we first select 50 images that the attribute is not present in. Then, we edit these images with a corresponding caption, such as *The person has blonde hair*. Finally, we use pre-trained attribute classifiers to measure the manipulation accuracy on the output images. We average the accuracy accross the attributes to obtain the final AMA score. We trained attribute classifiers for each of the 40 attributes that are present in the CelebA [Liu et al., 2015] dataset. We used the 15 attributes that achieve 90% or higher validation accuracies to calculate the AMA scores. Here is a full list of attributes we used for the CelebA dataset:

- *blonde hair*
- *double chin*
- *gray hair*
- *bushy eyebrows*
- *eyeglasses*
- *heavy makeup*
- *chubby*
- *goatee*
- *male*

- *mouth slightly open*
- *rosy cheeks*
- *wearing lipstick*
- *mustache*
- *smiling*
- *wearing necktie*

A.2 Additional Performance Comparisons

In Fig. A.1, we provide additional comparisons in few-shot domain adaptation against StyleGAN-NADA [Gal et al., 2022], RSSA [Xiao et al., 2022] and DynaGAN [Kim et al., 2022b]. Additionally, in Fig. A.2, we present additional comparisons of our approach against the BlendGAN [Liu et al., 2021a], TargetCLIP-O [Chefer et al., 2022] and TargetCLIP-E [Chefer et al., 2022] models in reference-guided editing. Finally, in Fig. A.3, we provide additional qualitative comparisons in text-driven manipulation against TediGAN-B [Xia et al., 2021], StyleCLIP-LO [Patashnik et al., 2021], StyleCLIP-GD [Patashnik et al., 2021], HairCLIP [Wei et al., 2022], DeltaEdit [Lyu et al., 2023], CLIPInverter [Hidden, 2023], DiffusionCLIP [Kim et al., 2022a] and plug-and-play [Tumanyan et al., 2022].

A.3 Editing Results for Other Domains

Our reference-based editing approach is not only limited to the faces domain. We trained our HyperCLIP-GAN on the CUB-Birds dataset [Wah et al., 2011] to demonstrate the generalization capabilities of our approach. When training these models, we use the same losses as described in the main paper except for the identity preservation loss, where we alternatively employ a ResNet50 [He et al., 2015] network trained with MOCOv2 [Chen et al., 2020]. In Fig. A.4, we provide various text-guided editing results, and in Fig. A.5, we provide several reference-guided synthesis results for the CUB-Birds dataset.

A.4 Δ-CLIP Space Analysis

For our text-based editing experiments, we observed that training on and using CLIP text embeddings directly degrades the manipulation quality. Instead, we employ the

Figure A.1: **Additional qualitative comparison against the state-of-the-art few-shot domain adaptation methods.** Our proposed HyperGAN-CLIP model outperforms competing methods in accurately capturing the visual characteristics of the target domains. The synthesized images exhibit a higher degree of fidelity and realism, demonstrating the effectiveness of our approach.

Δ-CLIP space where we simply take the difference of the target and source CLIP embeddings and use it as the conditioning. Fig. A.6 illustrates examples where we compare the results of text-based editing using the raw CLIP embeddings and Δ-CLIP embeddings.

Figure A.2: **Additional qualitative comparison with state-of-the-art reference-guided image synthesis approaches.** Our approach effectively transfers the style of the target image to the source image while effectively preserving identity compared to competing methods.

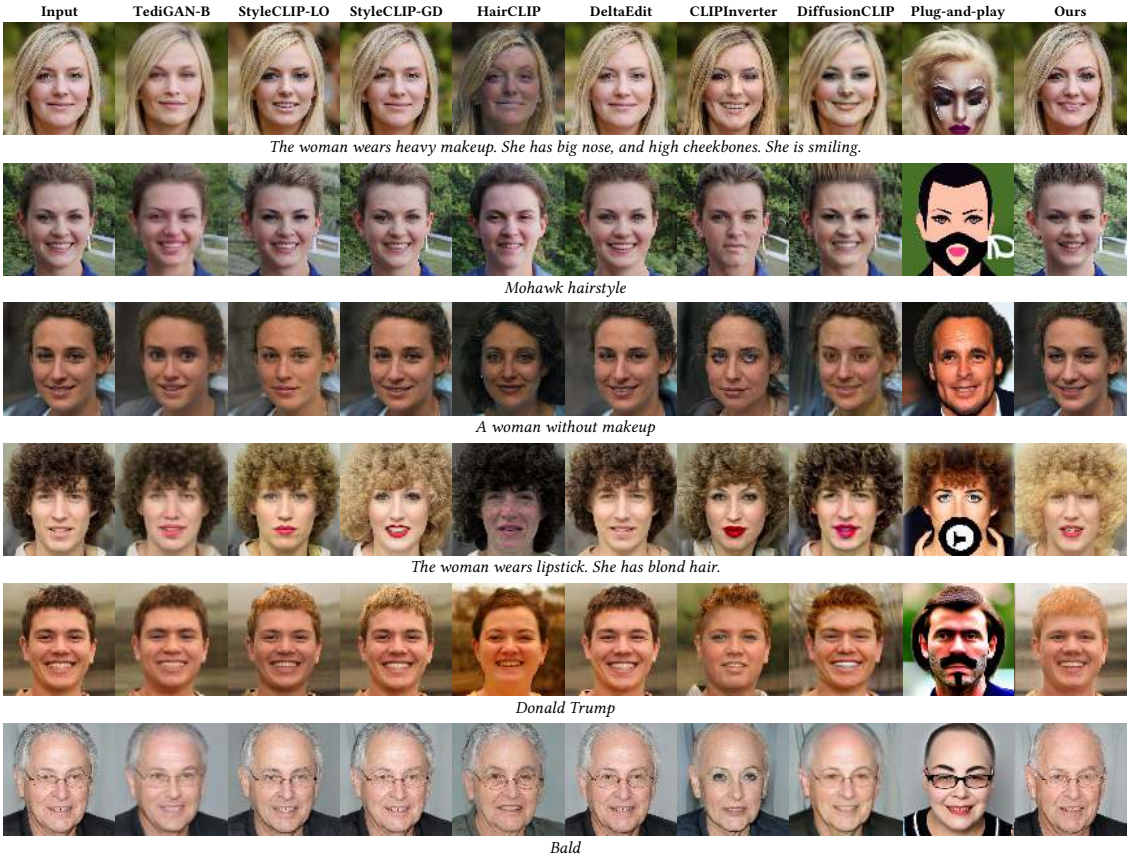

Figure A.3: **Additional qualitative comparisons with state-of-the-art text-guided image manipulation methods.** Our model shows remarkable versality in manipulating images across a diverse range of textual descriptions. The results vividly illustrate our model's ability to accurately apply changes based on target descriptions encompassing both single and multiple attributes. Compared to the competing approaches, our model preserves the identity of the input much better while successfully executing the desired manipulations.

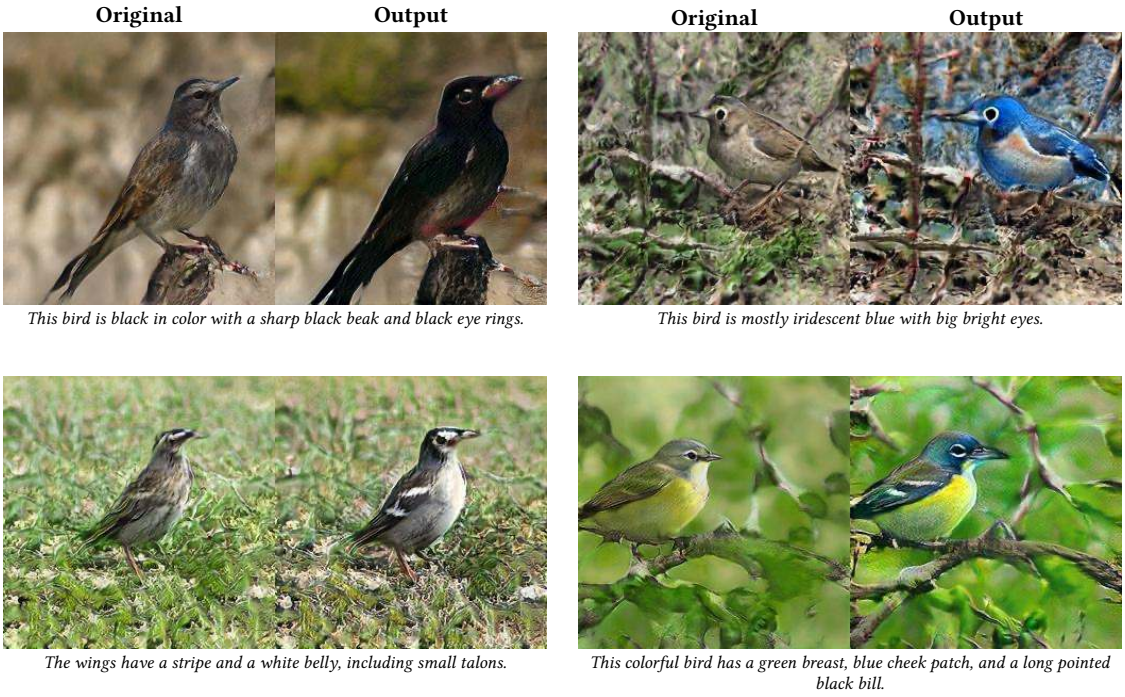

Figure A.4: **Text-guided editing results for the birds dataset.** Our approach generalizes to other domains, such as the bird images. We demonstrate zero-shot text-guided image editing results.

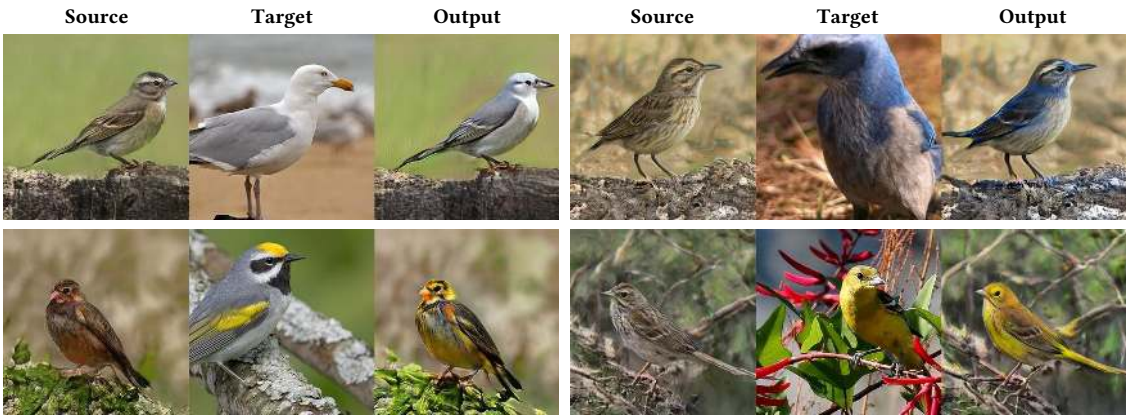

Figure A.5: **Reference-guided editing results for the birds dataset.** Our reference-guided synthesis generalizes to the birds domain, illustrated by the various targets we provide.

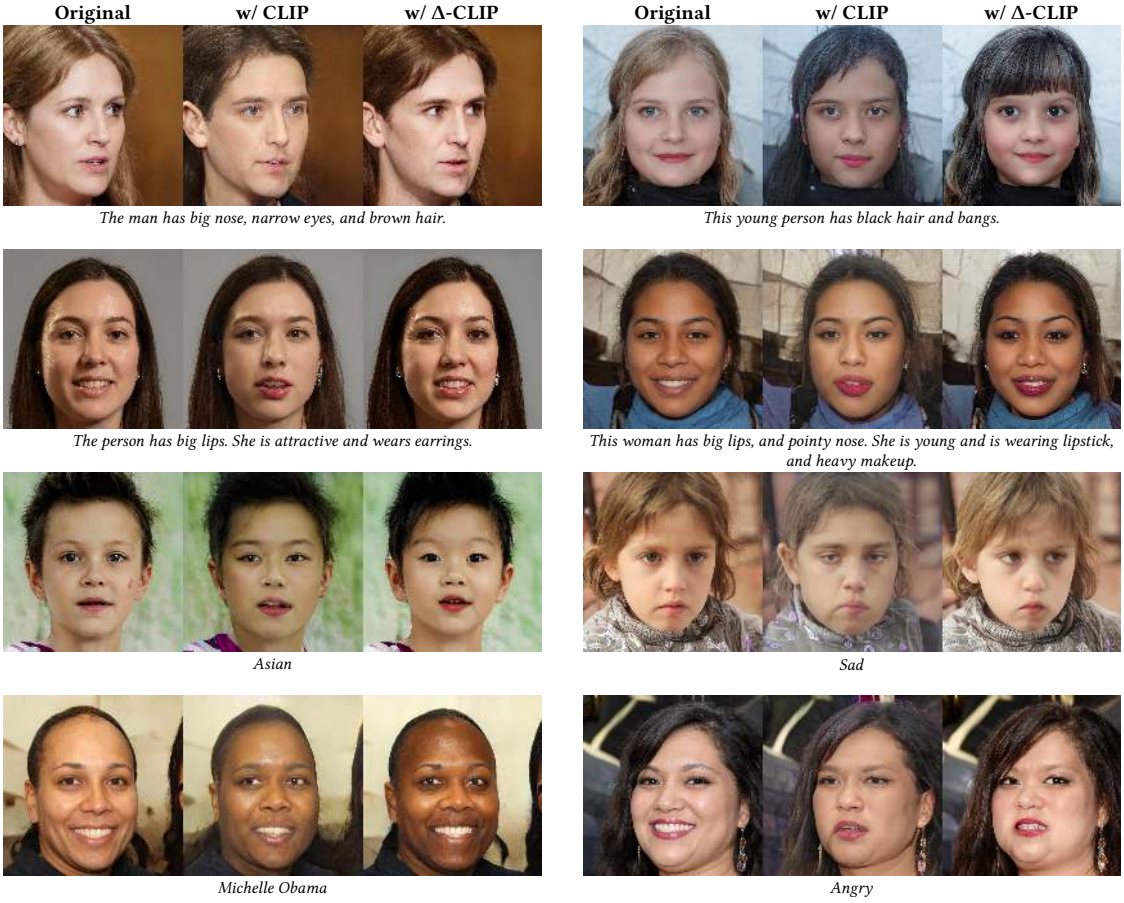

Figure A.6: **Text-based editing results using raw CLIP embeddings vs. Δ-CLIP embeddings.** As observed, we obtain much accurate manipulations while preserving the quality and fidelity when Δ-CLIP embeddings are used.