Vision Model Diagnosis and Improvement Via Large Pretrained Models

Yinong Wang

CMU-RI-TR-24-22

May 2024

School of Computer Science
The Robotics Institute

Carnegie Mellon University
Pittsburgh, Pennsylvania

Thesis Committee:

Fernando De la Torre, Chair
Jun-Yan Zhu
Nupur Kumari

Submitted in partial fulfillment of the requirements for the Degree of
Master of Science in Robotics

To my dear partner and parents.

Abstract

Recent years have witnessed a rapid evolution in the field of artificial intelligence (AI). As AI becomes increasingly pervasive in real-world applications, the deployment of machine learning models in real-world applications has underscored critical challenges in model robustness, fairness and performance. Despite significant advances, existing models often exhibit biases, fail to generalize across diverse data distributions, and struggle with unexpected input variations, leading to suboptimal or even discriminatory outcomes.

This thesis addresses these pressing challenges by harnessing the power of large pretrained models, especially vision generative models. In particular, two key problems are studied: (1) the identification of model biases and vulnerabilities, and (2) the utilization of synthetic data generation to improve model generalizability and performance. Along these lines, this thesis introduces two frameworks: Unsupervised Model Diagnosis (UMO) and Domain Gap Embeddings for Generative Dataset Augmentation (DoGE), which together offer a comprehensive and accessible solution to the challenges of model bias and distribution shifts in data.

UMO enables diagnosing model vulnerabilities in an unsupervised manner by employing generative models to produce semantic counterfactual explanations without the need for extensively annotated datasets or explicit user input. This framework facilitates the identification of sensitive semantic directions and spurious correlations within models, highlighting potential failure modes and biases without human intervention.

Complementing UMO, DoGE introduces a diffusion-based data augmentation technique that efficiently bridges cross-distribution gaps between training and target datasets. By capturing and embedding distribution differences in a latent form, DoGE enables the generation of synthetic datasets that closely align with target distributions, significantly improving model performance across various tasks.

The UMO framework's ability to diagnose model vulnerabilities without extensive annotated datasets or explicit user input, combined with DoGE's capability to augment data distributions to better align with target or underrepresented distributions, presents a powerful methodology for enhancing model fairness, robustness, and performance. Through these works, this thesis aims to enhance the robustness, fairness, and performance of machine learning models, thereby fostering the development of more reliable and equitable AI systems.

Acknowledgments

I am profoundly grateful to my advisor, Professor Fernando De la Torre, whose expertise, mentorship, and patience considerably shaped my graduate experience. You led me into the academic world and taught me crucial research and life philosophy as a graduate student. Your guidance helped me navigate through the challenges of research and the academic process with wisdom and grace. I sincerely appreciate your continuous, unreserved support and invaluable mentoring throughout this journey. I extend my sincere appreciation to the members of my thesis committee, Prof. Jun-Yan Zhu and Nupur Kumari for your time and insights in evaluating my work. I am grateful for your constructive feedback and suggestions.

I would like to thank all of my collaborators, Younjoon Chung, Eileen Li, Jinqi Luo, Chen Wu, Zhaoning Wang, and Prof. Fernando De la Torre for your hard work and insightful discussions that have contributed to the success of this thesis. Working with you has been both inspiring and encouraging as each one of you brought unique ideas and perspectives that have enlightened and enriched my research. I want to specially thank Jinqi Luo for handholding me through every step and detail of a complete research cycle and giving me a high-quality bar to strive for.

I am also immensely thankful to my peers and friends for all the great time we had together and the countless hours of discussions, brainstorming, and laughter that we shared, which provided me with the strength and support needed to overcome setbacks and power through the challenges of graduate school. I want to thank Younjoon Chung for being a great working and living buddy. I want to thank Iqui Balam, Christian Berger, Rwiddhi Chakraborty, Kwanghee Choi, Younjoon Chung, Prachi Garg, Vishnu Hema Mani, Hanzhe Hu, Kevin Joo, Jay Karhade, Kyurae Kim, Dorothy Ko, Viekash Vinoth Kumar, Bowen Li, Eileen Li, Yehonathan Litman, Ken Liu, Zhixuan Liu, Jinqi Luo, Aman Mehra, Diganta Misra, Sayan Mondal, Bharath Raj, Haoxi Ran, Aditya Rauniyar, Zhaoning Wang, Jiashun Wang, Jianren Wang,

Yake Wei, Chen Wu, Quanting Xie, Henry Xu, Jianjin Xu, Yu-Hsuan Yeh, Heng Yu, Cheng Zhang, Tianyuan Zhang, Runkai Zheng, and everyone else whom I am very lucky to meet here, for all the great time we had together. You have made my time at CMU truly memorable. I look forward to many more years of shared memories and successes.

Finally, my deepest gratitude goes to my partner Stella Chen and my parents who have provided me with unconditional love and support throughout my life and during the process of pursuing my graduate degree. Your belief in me always lifted my spirits and reminded me of the reasons why I embarked on this challenging journey. I am forever grateful for your unwavering support and encouragement. I love you all.

This thesis stands as a milestone in my academic journey, and I am thankful for everyone who played an invaluable role in shaping my journey thus far.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 AI - the Sword of Damocles

Throughout the history of technological evolution, the booming of artificial intelligence marks a revolution unlike any other, surpassing human competence in numerous exams [86], reshaping industries and ways of working [87], and even shedding light on our path to all-encompassing AI agents and systems. However, standing at this inflection point in 2024 and witnessing the explosive growth and integration of AI into people's daily lives, we cannot evade from confronting a critical fact: AI is the "sword of Damocles" of the 21st century.

The transformative power of AI is undeniable. With the emergence of unprecedented tools such as ChatGPT [11] and Stable Diffusion [102], AI has revolutionized countless industries and domains [81]. With the invention of video generation models like Sora [10], new possibilities for video content creation have emerged in the film industry [97]. Foundation models are becoming pivotal in advancing autonomous driving and robotics by enabling long-term reasoning and interaction with diverse agents, offering promising applications in these industries [132]. In healthcare, AI solutions are improving every aspect of patient care, from medical imagery and clinical studies to disease diagnosis and patient monitoring [3]. Yet, for all its transformative potential, the adoption of AI across these critical sectors is shadowed by concerns over its reliability, fairness, and transparency.

As AI systems increasingly make decisions that directly affect human lives, their trustworthiness becomes not just a technical consideration but also a societal imperative. Emphasized in the recent US President's executive order on AI safety, security,

and trustworthiness [116], this mandate reflects a growing recognition that AI systems must be fair, transparent, and reliable. The urgency of this issue is not merely academic; it is a foundational requirement for the continued integration and acceptance of AI in society. Without trust, the potential of AI to serve the greater good remains unrealized, hindered by legitimate concerns over bias, discrimination, and unintended consequences.

1.2 The Imperative of Trustworthiness in AI

However, building trustworthy AI systems faces challenges that are as complex as they are critical. Data, the cornerstone of AI, often carries the biases of its sources or the curation process, leading to models that inadvertently perpetuate or amplify these biases. The infamous instances of facial recognition technologies failing to accurately identify individuals of certain racial or ethnic backgrounds [34] highlight not just a failure of technology but a profound breach of ethical standards. Similarly, the susceptibility of autonomous driving systems to adversarial attacks [15] not only poses a safety risk but also erodes public confidence in the technology. These examples underscore the multifaceted nature of the challenge at hand, encompassing technical, ethical, and societal dimensions.

Moreover, the opacity of complex AI models further complicates the solutions for trustworthiness. The 'black box' nature of many deep learning systems, where the decision-making process and rationale are obscured and uninterpretable to humans, contrasts the demands for transparency and explainability. This lack of transparency not only impedes the ability of users to trust the system's decisions but also hinders efforts to diagnose and rectify biases or errors within the model.

1.3 A Comprehensive Approach for Model Diagnosis and Improvement

Addressing these challenges, this thesis proposes promising solutions toward trustworthy AI in computer vision from both the model and data aspect, leveraging the latest advancements in foundation models. Similar to doctors tending to patients by diagnosing and then treating the diseases, this work embarks on a two-phased journey: diagnosis and treatment of issues of computer vision models.

To treat model weakness and improve robustness, it is essential to first discover and understand the underlying biases and vulnerabilities. Indeed, relevant efforts have been abundant in the past from collecting challenging datasets like ImageNet-A [39] to adversarial attacks [78] and counterfactual generation [33]. However, observing traditional methods either rely on extensive test set collection or require unscalable human interpretation and intervention, the first half of this thesis focuses on a fully autonomous, unsupervised model diagnosis framework, Unsupervised Model Diagnosis (UMO), enabled by the latest advancements in generative models and large language models (LLMs). Given a target vision model, UMO leverages foundation toolkits to reveal the model's vulnerabilities and failure modes through visual counterfactual explanations and textual attribute analysis without any human supervision. This framework not only provides insights into the model's weaknesses but also guides further measures to enhance model robustness and fairness.

Building on this diagnostic foundation, the thesis transitions to its treatment phase, where the focus shifts to improving model generalizability and performance through synthetic data generation. Often, the training dataset can misalign with the actual production environment where the models are deployed. With the recent advancement of generative models, many works have studied synthetic dataset generation to improve model performance in the real world [67]. However, these prior works either rely on prompts that have limited expressiveness or require fine-tuning the generative model on desired distribution. Therefore, the second half of this thesis introduces Domain Gap Embeddings for Generative Dataset Augmentation (DoGE), a diffusion-based data augmentation technique that bridges the gap between training and test distributions without the need for explicit guidances or fine-tuning. By capturing and embedding distribution differences in a latent form, DoGE enables the generation of synthetic datasets that closely align with any desired data distribution (*e.g.*, the vulnerabilities discovered by UMO), significantly improving model performance across various tasks. It effectively 'vaccinates' AI systems against previous weaknesses, enhancing their fairness, robustness, and reliability.

The contribution of this thesis to the field of AI is multifaceted. The proposed two-phase pipeline encapsulates a novel approach to navigating the complexities of trustworthy AI development. Targeting the expensive and tedious past solutions to enhance model fairness and robustness, this thesis democratizes the process of discovering and addressing model vulnerabilities and makes the AI trustworthiness toolkit widely accessible to the community. By combining the power of generative

models and large language models, this work offers a comprehensive, autonomous, and scalable solution to the challenges of model bias, vulnerability, and distribution shifts.

In conclusion, the journey towards trustworthy AI is constantly accompanied by intricate challenges. However, through the meticulous diagnosis and targeted treatment of AI's vulnerabilities, this thesis illustrates a democratized path forward. By engraving the principles of fairness, trustworthiness, and robustness into AI systems, we can mitigate the risks and make good use of the modern "sword of Damocles". In doing so, we ensure that the future of AI is not just about technological advancement, but about advancing technology in harmony with the values and aspirations of society.

Chapter 2

Model Diagnosis

2.1 Introduction

Contemporary methods for assessing computer vision algorithms primarily rely on the evaluation of manually labeled test sets. However, relying solely on metric analysis of test sets does not ensure the robustness or fairness of algorithms in real-world scenarios [23]. This limitation arises from several factors. Firstly, while this approach is effective at gauging performance under known test conditions, it does not proactively address unforeseen model failures. Secondly, it is often infeasible to gather test sets that encompass all potential scenarios or relevant attributes. Lastly, the process of constructing test sets is typically resource-intensive, time-consuming, and susceptible to errors. To address such issues, the first half this thesis leverages large pre-trained models (LPMs), trained on extensive datasets comprising millions of samples, as a means to assess potential shortcomings in computer vision models. The question that we try to address is: *Can these LPMs be applied to evaluate various computer vision task models by uncovering possible failure modes and limitations in a completely* ***unsupervised*** *manner?*

An emergent approach to discover model failure modes without exhaustive test sets makes use of counterfactual explanations [80, 33]. Along this line of research, model defects are studied through challenging images that lead to model failure modes. However, earlier efforts deceive the target models via pixel-level adversarial perturbations [32, 78], which are not informative and do not explain model failures. Some works produce adversarial examples by semantically perturbing base images

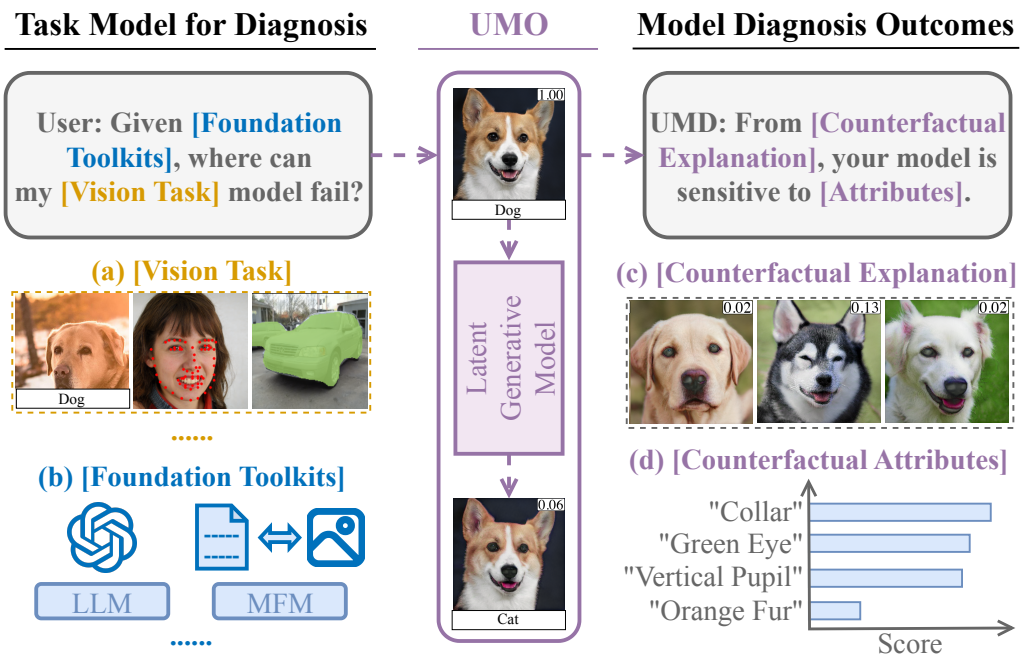

Figure 2.1: **Overview.** Given a (a) computer vision model (*e.g.*, classifier, keypoint detector, segmentation model), how can we understand the model vulnerabilities without requiring user input nor test sets? Our framework UMO leverages (b) foundation toolkits (*e.g.*, large language models (LLMs) and multi-modal foundation models (MFMs)) to perform **unsupervised** model diagnosis. UMO not only outputs (c) counterfactual visual explanations but also (d) top-matched counterfactual attributes.

along semantics [54, 95], but they focus on effective attacks rather than model evaluation. To gain insights into a model's flaws, we need counterfactual explanations that reveal the semantic differences leading to model failures. Recently, [75] proposed a zero-shot method to analyze model sensitivity to attributes via counterfactual explanation. Nonetheless, this method still requires human input, which can introduce bias and limit the outcome to the domain knowledge of particular users.

To address the issues mentioned above, we introduce Unsupervised Model Diagnosis (UMO) to discover the model failures and perform open-vocabulary model diagnosis without any user input or domain knowledge. Fig. 2.1 illustrates the main idea of our work. UMO comprises two main input components: (a) a target vision

task model provided by the user; in this thesis, we address classification, segmentation, and key-point detection, (b) a collection of foundation toolkits, *i.e.* LPMs, for counterfactual image generation as well as language models for semantic analysis. The output of UMO is a diagnostic report that includes (c) a set of visual counterfactual explanations and (d) the corresponding list of counterfactual image attributes with their associated semantic similarity scores.

The resulting diagnosis offers insights into the specific visual attributes to which the model is vulnerable. This information can guide actions such as collecting additional data for these attributes or adjusting their weights in the training process. Our complete pipeline operates in an unsupervised manner, eliminating the requirement for data collection and label annotation.

To summarize, our proposed work brings two main advancements:

- We propose an unsupervised framework for model diagnosis, named UMO, that bypasses the tedious and expensive requirement of human supervision.
- UMO utilizes the parametric knowledge from foundation models to ensure accurate analysis of model vulnerabilities. The framework does not require a manually-defined list of attributes to generate counterfactual examples.

2.2 Related Work

2.2.1 Latent Generative Models

Generative models, especially StyleGAN [58, 57, 107] and Diffusion Models [42, 114, 101], have semantic latent spaces that are differentiable, and can be used to edit image attributes [110, 48, 109]. StyleSpace [128] found a more disentangled latent space by manipulating the modulation weights (*i.e.*, style codes) in StyleGAN affine transformation. One step further, StyleCLIP [88] guided the generation sampled from StyleSpace by minimizing the CLIP [96] loss between the sampled image and the user text prompt. Recent advances in large language models enable informative supervision of the generation process. [140, 69] use language models to generate multi-modal conditions for enhanced compositionality and reasoning of diffusion models. In the line of controlling diffusion latent space, ControlNet [138] learned a task-specific network to impose constraints on Stable Diffusion [101]. Asymmetric reverse process (Asyrp) [64] discovered that the bottleneck layer of the U-Net in the diffusion model encodes meaningful semantics; hence, the authors proposed to learn a unified

semantic edit direction applicable across all images in this bottleneck latent space. Our work adopts the similar concept of optimizing globally effective counterfactual directions for the target model. By manipulating the semantics and analyzing the learned direction, we can gain valuable insights into model failure visualization and bias identification.

2.2.2 Diagnosis of Computer Vision Models

Model diagnostics [21] originally referred to the validity assessment of a regression model, including assumption exploration and structural examination. Recent years have witnessed the trend of the vision community broadly adopting the term *diagnose* [134, 75, 120, 93] for understanding and evaluating the failure of deep vision models, particularly focusing on attribution bias, adversarial robustness, and decision interpretability to identify potential flaws. To search model failure cases, [32] first proposed pixel-space perturbations by signed gradient ascent to generate adversarial examples. [78, 130] further advanced the philosophy by multi-step gradient projection and incorporating generative models.

However, [54, 95] claimed that such adversaries lack visual interpretability and proposed attacking the model by optimizing along fixed semantic axes of generative models. Similarly, in the literature of counterfactual explanation [122], methods [51, 135, 61, 1, 49, 113, 80] commonly focus on generating semantic attacks on a per-image basis, which overlook the global model-centric vulnerability diagnosis. Despite the effective instance-level counterfactual generation, such failure-driven attacks can be less informative for diagnosing model vulnerabilities (*e.g.*, altering the perceived gender to fool a gender classifier). Hence, when directly applying these attack-oriented counterfactual explanations for model diagnosis, they usually require additional human interpretation to summarize individual failures. Besides being often designed for specific single task [92, 135, 49], previous methods [100, 1, 113] also require fine-tuning of the generative pipeline. Hence, we adopt a diagnosis-driven, task-agnostic, and resource-efficient counterfactual synthesis pipeline desirable for diagnosing instead of simply attacking models.

To emphasize the explainability requirement of model diagnosis and produce human-understandable outcomes, recent works visualized model failures by optimizing an attribute hyperplane [68], identifying error subgroups from cross-modality gaps [134, 25], recognizing sensitive style coordinates [65], searching semantic variations by

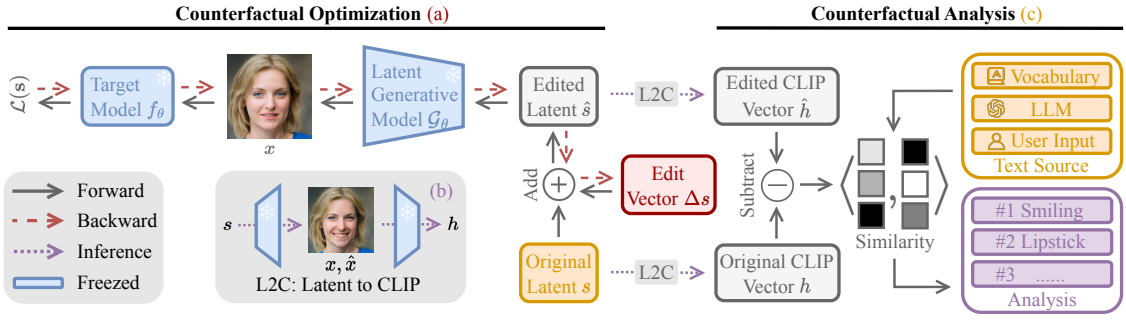

Figure 2.2: **The UMO framework.** Black solid lines denote forward passes; red dashed lines denote backpropagation; and purple dotted lines denote the inference of analysis. (a) We first optimize an edit direction Δs in the latent space of generative models that yields counterfactual images of the target model. (b) After the optimization converges, we generate the original and edited images x and $\hat{x}$ and map them to the CLIP embedding space with the L2C block (in **??**). (c) Then we analyze and report the diagnosis of counterfactual attributes by matching the image embedding differences $\hat{h} - h$ with attribute candidates.

unconditional generative models [66, 52], or fine-tuning language model to perturb prompts for text-condition generations [93]. Nevertheless, these approaches require either manually annotating the discovered failure and the collected test set, or training a model-specific explanation space for every new target model. Our approach addresses these shortcomings by performing diagnosis in an unsupervised manner with the help of foundation toolkits. More recently, ZOOM [75] proposed to analyze provided attributes in a zero-shot manner. This is the most relevant method to UMO to the best of our knowledge. However, our method distinguishes itself from ZOOM in that: (1) UMO achieves automatic discovery instead of focusing on analyzing given attributes, (2) UMO is more exploratory in counterfactual edits while ZOOM can only analyze in restricted attribute directions, (3) UMO requires no prior knowledge from the user and hence circumvents potential human biases.

2.3 Method

Given a target model f_θ, our pipeline consists of two stages as shown in Fig. 2.2. In the first stage, a latent generative model, denoted as $\mathcal{G}_\phi$, (*e.g.*, a Diffusion model or GAN) is used to discover counterfactual modifications, denoted as UMO$(f_\theta, \mathcal{G}_\phi)$, by directly optimizing the latent edit directions that can mislead the prediction of f_θ, shown in

Fig. 2.2(a). In the second stage, after the counterfactual optimization converges, we generate pairs of image embeddings (original, counterfactual) in Fig. 2.2(b). With these pairs, we are able to interpret and analyze the counterfactual attributes of the target model f_θ by computing semantic similarity scores with attribute candidates, as illustrated in Fig. 2.2(c).

2.3.1 Counterfactual Optimization

Given a target model f_θ to diagnose, we first focus on discovering cases that lead to incorrect model predictions. To capture the failure modes, an effective solution is to generate counterfactual examples of the target model. Hence, our first step aims to learn latent edits which represent semantic edit directions. When injecting such latent modification to the generative model $\mathcal{G}_\phi$, the generated images are observed to have meaningful semantic changes but are challenging to the target model. We denote each pair of the original image x_i and its edited adversarial counterpart $\hat{x}_i$ as a counterfactual pair. This section shows that UMO can discover and generate critical semantic counterfactual pairs of target model f_θ using various generative models $\mathcal{G}_\phi$ (*e.g.*, StyleGAN and Diffusion Model).

Since the StyleSpace $\mathcal{S}$ is shown to be effective for semantic manipulation [128, 88], we choose to inject counterfactual edits in this latent space. We first initialize our latent edit vector Δs in the space $\mathcal{S}$ as Gaussian noise. Then we sample N style vectors $\{s_i\}_{i=1}^N \sim \mathcal{S}$ and generate N corresponding images $x_i = \mathcal{G}_\phi(s_i)$. Note that optionally real images x_i can also be used and s_i is then obtained through GAN inversion[129]; but for the rest of this work, we use synthetic images x_i to leverage free exploration in generative latent space for more diverse counterfactual explanations. For each original image x_i, we inject the edit vector Δs to the latent space $\mathcal{S}$ and obtain the edited image $\hat{x}_i = \mathcal{G}_\phi(s_i + \Delta s)$. With the original and edited image based on the same initial latent vector, we compute the following loss $\mathcal{L}$:

$$\mathcal{L}(\Delta s) = \alpha \mathcal{L}_{\text{target}} + \beta \mathcal{L}_{\text{CLIP}} + \gamma \mathcal{L}_{\text{SSIM}} + \mathcal{L}_{\text{reg}}. \tag{2.1}$$

To optimize w.r.t. Δs, we back-propagate the loss as shown in the backward process in Fig. 2.2(a).

The first component of the loss, $\mathcal{L}_{\text{target}}$, is the adversarial loss that ensures the learned Δs can edit the original image to cause target model failure. This loss measures the distance between the model prediction on the edited image $f_\theta(\hat{x}_i)$ and the

opposite of the original model prediction $f_\theta(x_i)$. It is task-dependent: when diagnosing a binary classification task, we want to minimize the cross-entropy loss between $f_\theta(\hat{x}_i)$ and $\hat{p}_{\text{cls}} = 1 - f_\theta(x_i)$ such that the edits effectively mislead the classifier toward the opposite class; when diagnosing a keypoint detector or a segmentation model, we want the (perturbed) incorrect model predictions to be close to a randomized or targeted pseudo label (details in Sec. 2.4.3), denoted as $\hat{p}_{\text{kdet}}$ and $\hat{p}_{\text{seg}}$ respectively:

$$\text{(binary classifier) } \mathcal{L}_{\text{target}}(\hat{x}_i) = L_{\text{CE}}(f_\theta(\hat{x}_i), \hat{p}_{\text{cls}}), \tag{2.2}$$

$$\text{(keypoint detector) } \mathcal{L}_{\text{target}}(\hat{x}_i) = L_{\text{MSE}}(f_\theta(\hat{x}_i), \hat{p}_{\text{kdet}}), \tag{2.3}$$

$$\text{(segmentation model) } \mathcal{L}_{\text{target}}(\hat{x}_i) = L_{\text{CE}}(f_\theta(\hat{x}_i), \hat{p}_{\text{seg}}). \tag{2.4}$$

Optimizing against $\mathcal{L}_{\text{target}}$ solely without constraints is insufficient to discover effective counterfactual examples. For example, with a binary classifier, the original image can be directly edited into the opposite class [51, 135, 61, 1, 113] which fails to reveal the failure modes of the target model. Hence we introduce the second loss term $\mathcal{L}_{\text{CLIP}}$ which ensures the generated counterfactual example is perceived by the CLIP model as the same class/object as the unedited image. Denoting the CLIP zero-shot classifier as $\mathcal{C}$ and the list of class labels as $\mathcal{T}$:

$$\mathcal{L}_{\text{CLIP}}(\hat{x}_i) = L_{\text{CE}}(\mathcal{C}(x_i, \mathcal{T}), \mathcal{C}(\hat{x}_i, \mathcal{T})). \tag{2.5}$$

While optimizing $\mathcal{L}_{\text{CLIP}}$ and $\mathcal{L}_{\text{target}}$ yields counterfactual examples to the target model, we also preserve the quality of the counterfactual by regularizing attribute changes and preserving semantic structures. To enforce these constraints, we include the SSIM loss [124] and the regularization as:

$$\mathcal{L}_{\text{SSIM}}(\hat{x}_i) = L_{\text{SSIM}}(x_i, \hat{x}_i), \tag{2.6}$$

$$\mathcal{L}_{\text{reg}}(\Delta s) = ||\Delta s||_1. \tag{2.7}$$

To mitigate any implicit bias inherited from one specific generative backbone, we further enhance the reliability of our diagnosis pipeline through an ensemble of latent generative models. We propose to generate counterfactual images from multiple independent generative backbones and analyze the combined mixture of synthesized images. Hence, besides StyleGAN, we also adopt diffusion models for counterfactual generation. Asyrp [64] discovers a semantic latent space in diffusion models, in

particular, DDPM [42] and DDIM [114]. Asyrp proposes to learn a simple network A that takes the hidden states from a previously denoised image x_t at the timestep t and outputs an edit vector to be injected to the middle bottleneck layer of each U-Net block. Similar to optimizing Δs in the StyleSpace of StyleGAN, we optimize the network A in diffusion models to learn counterfactual edits of the target model.

Since the decision behavior of the target model can be biased toward multiple attributes, we choose to optimize k distinct edits to ensure comprehensive diagnosis coverage and to improve optimization convergence by focusing each vector on one type of edit. We initialize k latent edit vectors for StyleGAN or k edit generation networks for diffusion models. Then, for each original latent vector, we first find the edit vector that most effectively perturbs the target model and only optimize this edit vector while leaving the remaining edit vectors unchanged. This procedure repeats at each iteration. Since distinct failure modes can emerge for different latent vectors, each edit vector converges to different and disentangled semantic edit directions throughout training, which enables the discovery of multiple biased attributes. More details are shown in Appendix A.1.

2.3.2 Counterfactual Analysis

To render an intuitive diagnosis of the target model, we interpret the attribute changes between the counterfactual pairs that lead to target model failure. Using the CLIP model as a common latent space, we match the counterfactual edits with text attribute candidates to provide analyses of the model vulnerabilities.

After counterfactual optimization, we augment original latent vectors s into $\hat{s}$ by adding the learned edit vector Δs. Then we feed both s and $\hat{s}$ into the Latent-to-CLIP (L2C) module. L2C generates two images x and $\hat{x}$ from s and $\hat{s}$ and encode them into the CLIP embedding space, as depicted in Fig. 2.2(b). To illustrate, consider an original picture x of a woman correctly classified as "female" by the target model, but a smile is added in $\hat{x}$, and the classifier now incorrectly predicts "male". To extract the differences between the pair of images, we use a pretrained CLIP image encoder $\mathcal{E}_I$ and convert each counterfactual pair $(x, \hat{x})$ to a CLIP embedding pair $(h = \mathcal{E}_I(x), \hat{h} = \mathcal{E}_I(\hat{x}))$. We then extract the image difference in the CLIP space as $\Delta h = \hat{h} - h$.

To interpret Δh and further diagnose the target models, we match Δh to a repository of text attribute candidates and report the top-n ranked text attributes. There

can be many sources of these candidates: the entire vocabulary of the Brown Corpus [26] can be used; we can also prompt a language model to provide a shorter but extensive list of all relevant attributes; if desired, users can also provide a list of particular attributes of their interests. For efficiency and concision, we use language models as our bank of attribute candidates for UMO. See Appendix A.2 for the details of attribute candidate generation.

We denote the set of attribute candidates as $S_a = \{a_i\}_{i=1}^M$ and the known object of focus as `[cls]`. For each attribute a_i, we prepare the pairs of base prompt and attribute prompt as t_{base}: "an image of `[cls]`" and t_i: "an image of `[cls]`, `[`a_i`]`" respectively. Using an off-the-shelf CLIP text encoder $\mathcal{E}_T$, we extract the prompt difference as $\Delta t_i = \mathcal{E}_T(t_i) - \mathcal{E}_T(t_{\text{base}})$. Then the similarity score for attribute a_i is defined as:

$$S_{\text{sim}}(a_i) = \mathbb{E}_{G_\phi \sim p(G_\phi)}[\mathbb{E}_{x \sim G_\phi(x)}[\langle \Delta h, \Delta t_i \rangle]], \tag{2.8}$$

where $p(G_\phi)$ denotes the set of generative models.

We select the j highest-ranked attributes by the similarity score into our diagnosis. Since the attribute candidate bank can be repetitive, the top-j selected candidates can be dominated by few related attributes. Thus, we also introduce a uniqueness score to encourage matching with distinct new attributes in an iterative fashion. Let the set of already selected attributes be S_r, initialized as an empty set, the uniqueness score is defined as:

$$S_{\text{uni}}(a_i) = \begin{cases} 1, \text{ if } S_r = \emptyset \\ 1 - \max\limits_{t \in S_r} \langle \mathcal{E}_T(a_i), \mathcal{E}_T(t) \rangle, \text{otherwise.} \end{cases} \tag{2.9}$$

At each iteration, the next highest-ranked attribute is selected into S_r as:

$$S_r = S_r \cup \{\underset{a_i \in S_a}{\text{argmax}}\ S_{\text{uni}}(a_i) \cdot S_{\text{sim}}(a_i)\}. \tag{2.10}$$

By iteratively repeating the attribute interpretation process across all counterfactual pairs, we obtain the top matching counterfactual attributes that can result in model prediction failures, see Appendix A.3 for details.

2.3.3 Counterfactual Training

While the counterfactual pairs $(x, \hat{x})$ offer visual insights into the vulnerabilities of the target models, the counterfactual images $\hat{x}$ can also directly serve as the hard training set for fine-tuning target models. This section adopts the principle of iterative adversarial training [78] on these generated counterfactual images to fine-tune the target models.

We start with the pre-trained target model. At each iteration, we optimize and generate a set of counterfactual images with respect to the current model state and concatenate them with regular training data of the same size. This way, we dynamically improve model robustness at each training step. Compared to ZOOM, in which generated counterfactual examples are constrained along fixed attributes, UMO can dynamically adapt the counterfactual directions depending on the current state of the model during training. This training process is essentially a min-max game where UMO keeps searching for new weaknesses and the target model is subsequently patching it. UMO iteratively enhances counterfactual robustness of the target model in an unsupervised fashion. Appendix A.4 shows the effect and robustness improvement of our counterfactual training.

2.4 Experimental Results

This section presents the experimental results evaluating the validity and effectiveness of UMO. We first verify the correctness of our diagnosis in Sec. 2.4.1. Then Sec. 2.4.2 demonstrates the consistency of our diagnosis across different generative models and with a prior method. Sec. 2.4.3 further illustrates the broad applicability of UMO to more computer vision tasks. An ablation study of the effect of each loss component is shown in Sec. 2.4.4. Lastly, Sec. 2.4.5 assesses the validity of foundation toolkits as the backbones of our diagnosis task. All our experiments are done on a single Nvidia RTX A4000 GPU with 16GB of memory. The hyperparameters α, β, and γ that correspond to the weights of the target, CLIP and SSIM losses are tuned empirically to be $\alpha = 1$, $\beta = 10$ and $\gamma = 100$.

2.4.1 Diagnosis Validation with Imbalanced Data

This section evaluates UMO through experiments with classifiers trained on imbalanced data. It is important to note that no definitive ground truth exists in this

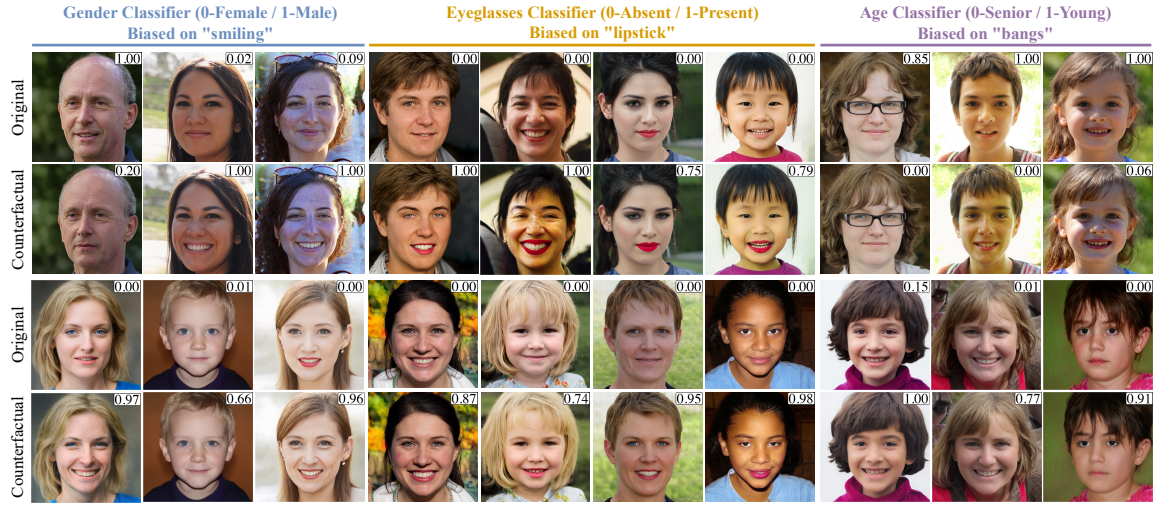

Figure 2.3: **Counterfactual pairs generated against different classifiers.** We study a perceived gender classifier biased on "smiling" (left), an eyeglasses classifier biased on "lipstick" (middle), and a perceived age classifier biased on "bangs" (right). For each classifier, we optimize the semantic latent edits to obtain counterfactual variations (bottom row) from the original generations (top row). This figure demonstrates the capability to provide visual counterfactual explanations on the biases of these classifiers.

setting. We carefully constructed target models embedded with specific biases to serve as our reference (*i.e.*, ground truth). Our experiments consistently highlight that the diagnosis from UMO can reliably pinpoint these intentional biases, confirming the reliability of our unsupervised detection.

In this set of experiments, we examined classifiers that were intentionally biased and trained using the CelebA dataset [70]. We trained binary classifiers for specific attributes within CelebA, such as "gender", "age", and "eyeglasses". For each classifier, we chose another secondary attribute to introduce artificial (spurious) correlations, achieved through imbalanced sampling. For instance, when assessing a perceived gender classifier biased by the presence of smiles, we curated a subset from the CelebA dataset, containing 10000 males with smiles, 10000 females without smiles, and 100 images with the opposite smile presence per class. A perceived gender classifier was trained on this subset, producing a model with a known bias on "smiling". We repeated this procedure to produce a set of attribute classifiers with known biases.

On these classifiers, if UMO can successfully discover and report the planted

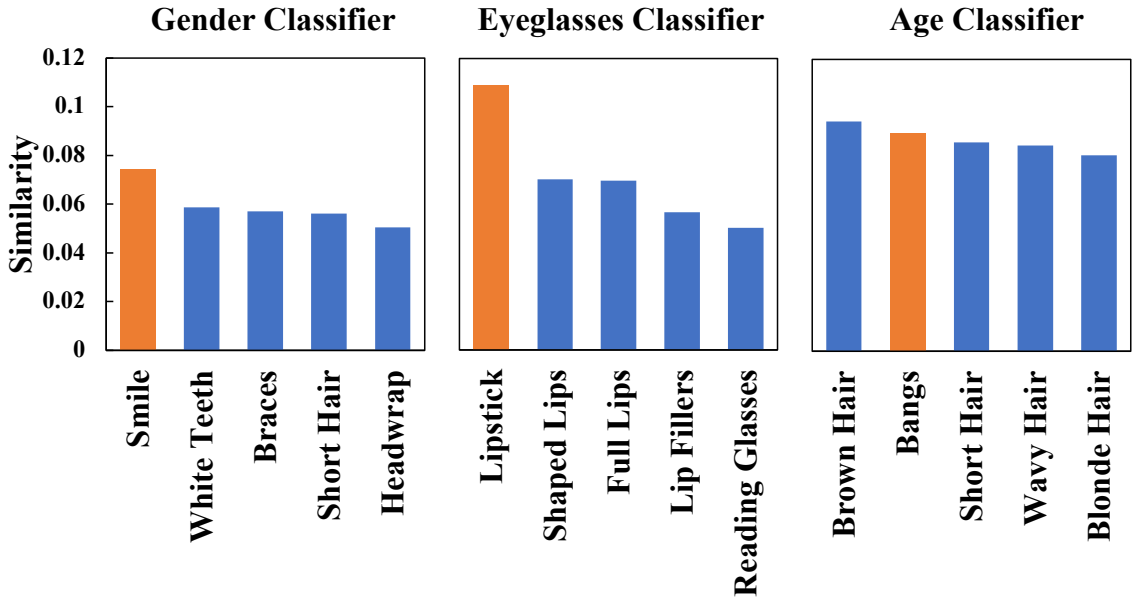

Figure 2.4: **Top-5 discovered attributes and their similarity scores, with the planted bias highlighted in orange.** For a given target classifier, the similarity score of each attribute is computed through the counterfactual analysis module. These experiments indicate that our unsupervised diagnosis pipeline is indeed capable of discovering the bias in a given model.

biases, then we can verify the validity and effectiveness of our pipeline. Fig. 2.3 shows the visual explanations from the unsupervised counterfactual optimization with both the StyleGAN and DDPM backbones for each of the three CelebA classifiers. Besides the visualization, UMO analyzed 1000 such generated counterfactual pairs of original and edited images. The top-five discovered attributes are shown in Fig. 2.4. For the perceived gender and eyeglasses classifiers, as expected, "smiling" (left) and "lipstick" (middle) are discovered as the top attributes with a significant margin. In the Age classifier, the planted attribute "bangs" (right) surprisingly was second-ranked in the analysis after "brown hair". However, upon a close look in CelebA, we found that 85.82% "brown hair" faces are labeled as "young". As so, this is a strong spurious correlation between "age" and "brown hair" uncovered by UMO. This experiment verifies that our pipeline can correctly discover both the planted and existing biases in the target model.

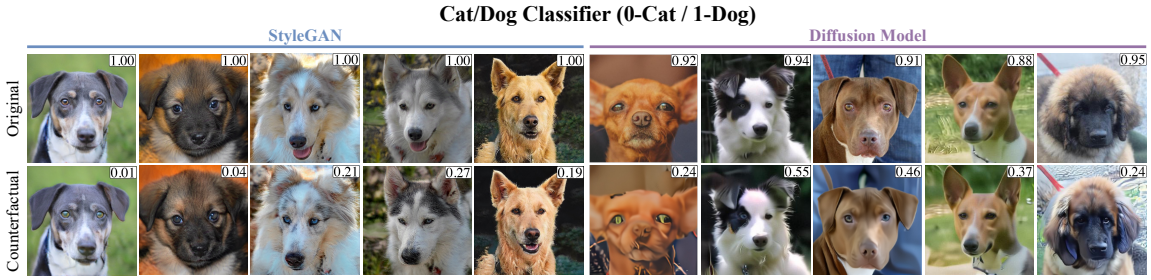

Figure 2.5: **Counterfactual pairs generated with different backbones.** We diagnose a Cat/Dog classifier and show visual counterfactual explanations from different generative backbones before the ensemble analysis. (left) and (right) are counterfactual pairs generated from StyleGAN and Diffusion models respectively. We can see both models make some common perturbations, most notably eye color changes.

2.4.2 Cross-Method Diagnosis Consistency

We further validate the effectiveness of our approach across different models, including a prior work, ZOOM. First, we trained a conventional Cat/Dog classification model on the AFHQ dataset [17]. Subsequently, we performed three distinct experiments: two using our framework with different generative model backbones (Diffusion model and StyleGAN), and the other using the ZOOM approach requiring a user-provided attribute candidates list. We should expect to discover the same counterfactual attributes for a given target model, across all three experiments.

Fig. 2.6 illustrates the consistency of counterfactual analysis conducted by UMO, irrespective of whether we utilize the Diffusion or StyleGAN generative model. In both of these experiments, the top six attributes were consistently ranked from a pool of 88 relevant attributes, which were automatically generated using our foundational toolkit. Specifically, attributes related to eye color (green or heterochromia), vertical pupil shape, dark fur color, and pointed ears emerged as counterfactual in all three methods. Furthermore, both StyleGAN and Diffusion backbones identified long whiskers and wearing a collar as additional attributes. This quantitative assessment of attribute rankings aligns with our qualitative observations in Fig. 2.5.

This consistency highlights the robustness of our diagnostic approach, regardless of the choice of generative backbones, as long as these choices provide a sufficiently rich semantic latent space. Furthermore, the significant counterfactual attributes we identified align with those found in previous research, specifically in ZOOM. It is important to note that the diagnosis process in ZOOM relies on human input, whereas

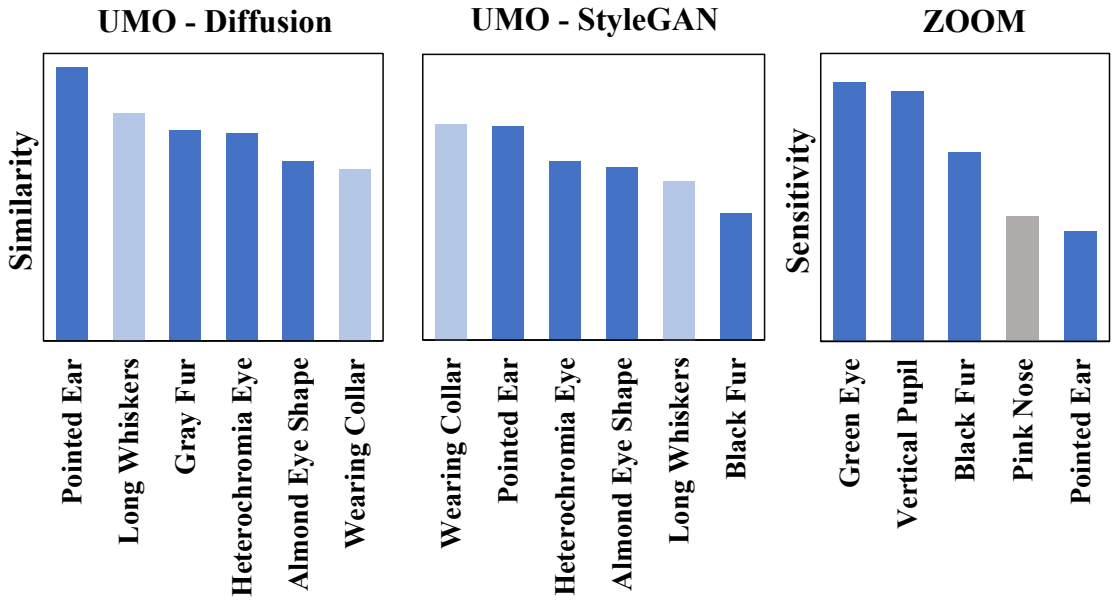

Figure 2.6: **Discovered attributes consistent across two backbones and one prior work against the same Cat/Dog classifier.** Here we performed counterfactual analysis separately on the generated counterfactual pairs from StyleGAN and Diffusion Model. We also include an analysis based on ZOOM. Dark and light blue attributes are respectively consistent across all three diagnoses and the two backbones in our framework. We observe consistency in the discovered attributes despite the generative backbone and method differences.

our unsupervised method allows for a more comprehensive analysis. As a result, we contend that our approach represents a generalization of ZOOM. It not only overcomes the limitations associated with user-proposed attributes but also circumvents biases stemming from user input, thus expanding the diagnostic capabilities across various generative models.

2.4.3 Generalization to Other Vision Tasks

In addition to classification, we expanded our experiments to encompass image segmentation and keypoint detection tasks. This extension demonstrates the versatility and practicality of UMO. We conducted diagnostics on a publicly available segmentation model trained on ImageNet [22] and a keypoint detector trained on the FITYMI dataset [125].

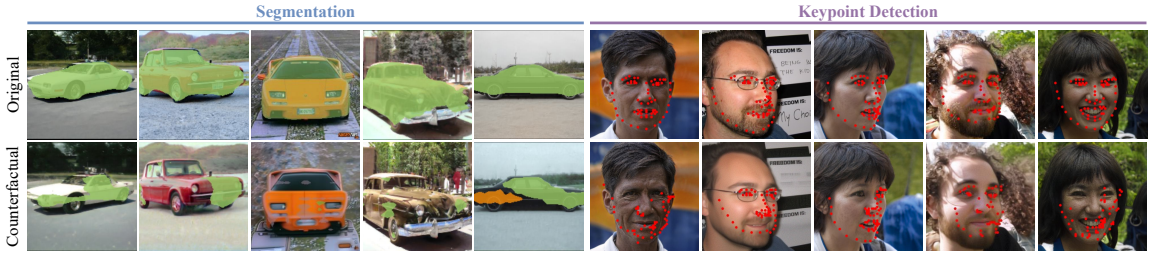

Figure 2.7: **Visual diagnosis on more computer vision tasks.** We applied UMO to two more computer vision tasks: (left) segmentation and (right) keypoint detection. Our pipeline successfully demonstrates semantic changes that fool the target model.

Segmentation	
Attribute a_i	S_{sim}
dirt road	0.0846
potholes/roadworks	0.0786
snow-covered road	0.0654

Keypoint Detection	
Attribute a_i	S_{sim}
beard	0.0947
elderly	0.0754
missing teeth	0.0728

Table 2.1: **Top-3 attributes diagnosed by UMO.** We apply our counterfactual analysis module in the segmentation and keypoint detection tasks and show the top three attributes diagnosed for each model. The discovered attributes reflect our observations in Fig. 2.7.

Similar to inverting the binary classification label, we establish a definition for $\hat{p}_{\text{kdet}}$ and $\hat{p}_{\text{seg}}$ in $\mathcal{L}_{\text{target}}$, the pseudo ground truth described in Sec. 2.3.1, to guide our counterfactual optimization in both tasks. In the segmentation task, our framework directs the target model to predict a suboptimal class (*i.e.*, the second most probable class) per pixel, as opposed to the most probable one. Similarly in the keypoint detection task, we attack the model by optimizing for random transformations of ground truth keypoints.

Fig. 2.7 illustrates counterfactual explanations for segmentation and keypoint detection. In both cases, our approach successfully uncovered semantic edit directions that deceive our target models. In segmentation, we observed that attributes related to road conditions, such as "snow-covered road" and "off-road" appear to influence model predictions. Conversely, in keypoint detection, we found that characteristics related to age, such as "beard", "wrinkles", and "freckled skin" play a crucial role in creating counterfactual instances. We list the top attributes ranked by the semantic similarity score S_{sim} in Sec. 2.4.3.

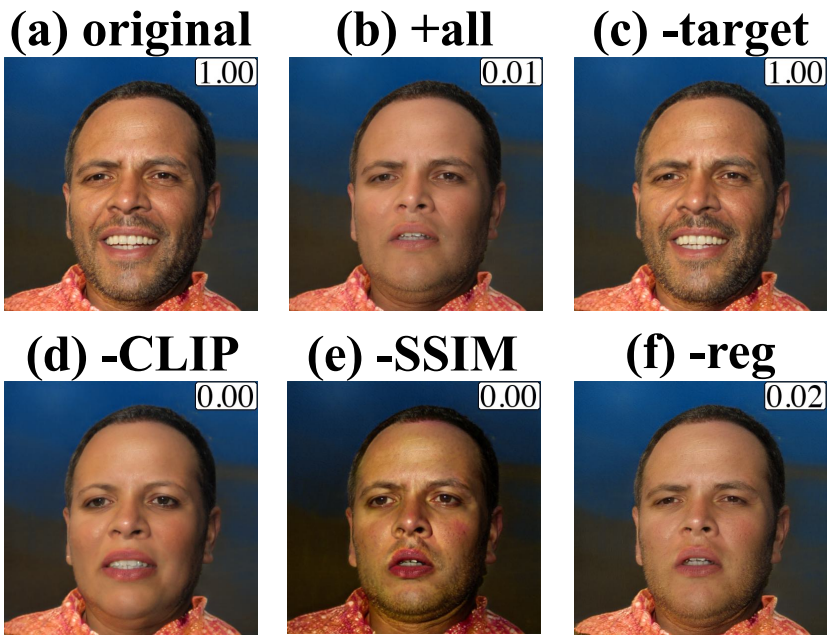

Figure 2.8: **Effect of removing each loss.** We optimize the counterfactual edit vector on the same gender classifier (0-Female / 1-Male) biased on "smiling" as in Sec. 2.4.1. Column (a) and (b) are the original and regular counterfactual images. Column (c)-(f) shows the different effects in counterfactual images from removing one loss component while keeping the other three.

2.4.4 Ablation Study of Loss Components

This section analyzes the contribution of each loss component in Eq. (2.1) in the counterfactual edit vector optimization. In this experiment, we reuse the setup from Sec. 2.4.1 and focus on the same perceived gender classifier with the attribute "smiling" planted purposefully as bias. We then ablate the loss component one at a time and generate counterfactual images with the same pipeline to visualize the isolated effect of each loss.

Fig. 2.8 shows an ablation analysis of the impact of each loss component. Fig. 2.8(a) shows the original unedited image which is correctly predicted by the classifier as perceived male. Fig. 2.8(b) presents the regular counterfactual image optimized with all loss components, which successfully flipped the classifier prediction by learning to remove the smile instead of altering the perceived gender. When ablating $\mathcal{L}_{\text{target}}$ in Fig. 2.8(c), we can see that the edit vector is hardly modifying the image since $\mathcal{L}_{\text{target}}$

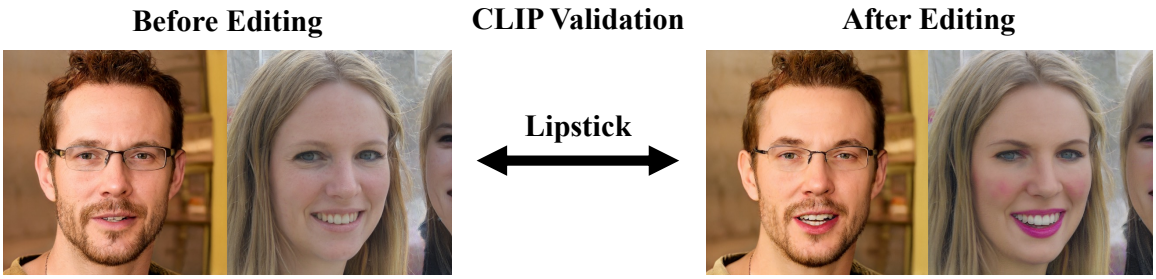

Figure 2.9: **Pairs of images randomly selected to validate CLIP as analysis backbone.** Between left and right, the most salient change is lipstick.

dictates the adversarial part in the optimization. Removing $\mathcal{L}_{\text{CLIP}}$ in Fig. 2.8(d) leads to the edit vector learning the "easiest" change which is simply flipping the target class (*i.e.* perceived gender). It shows the importance of our CLIP loss in producing informative and useful counterfactual examples for model diagnosis. Finally, the absence of $\mathcal{L}_{\text{SSIM}}$ and $\mathcal{L}_{\text{reg}}$ resulted in lack of the proximity (*e.g.*, different contrast in Fig. 2.8(e) and unnecessary edits around eyes in Fig. 2.8(f)), which interferes with the subsequent diagnosis.

2.4.5 Foundation Toolkit Validation

The effectiveness of UMO's diagnosis hinges on the capabilities and reliability of the foundational toolkits we incorporate. In this section, we present evaluations that underscore the dependability of both GPT-4 [86] and CLIP.

As highlighted in Sec. 2.3.2, we choose GPT-4 to serve as the primary source of attribute candidates within UMO. We task GPT-4 with producing comprehensive lists of attributes for each task domain, encompassing attribute types and their corresponding attribute values. In order to illustrate the effectiveness of GPT-4, we selected a set of representative attributes and provided a detailed breakdown of all the values associated with each attribute, as generated by GPT-4, in Sec. 2.4.5. This table offers a qualitative glimpse into the extensive capacity of GPT-4 in populating attribute candidates. For prompting details and the complete list of generated attributes, see Appendix A.2.

We evaluate CLIP's capabilities in providing text analysis from counterfactual images by examining the model's robustness and reliability when matching visual

Attributes	**Values**
Hairstyles	short, long, curly, straight, wavy, braided, bald, mohawk, bun, pixie cut, dreadlocks, undercut, pompadour, buzz cut, side part, bob cut, cornrows, bangs
Eye Colors	blue, brown, green, gray, hazel, black, amber
Nose Shapes	Roman, snub, Greek, aquiline, hawk, button
Expressions	smile, frowning, surprised, angry, crying, wink
Glasses Types	reading glasses, sunglasses, aviator, cat-eye, round, square, rimless glasses
Accessories	earrings, necklace, hat, cap, headscarf, headband, bandana, tie, piercing, bow tie
Background	indoor, outdoor, simple, busy

Table 2.2: **Examples of attribute candidates proposed by GPT-4.** The full candidate list and prompts used are in Appendix A.2. This list illustrates the comprehensiveness of large language models as attribute generators.

Injected Ambiguous Attributes
wrinkles, dimples, lip ring, lip piercing, lip fillers, lip gloss, bleeding lips, wounded lips, red face, red mustache, red hair, makeup, blush, contour, crooked teeth

(a) The list of ambiguous attribute candidates to attack our counterfactual analysis module.

Attribute a_i	S_{sim}
lipstick	0.1270
full lips	0.1106
lip gloss	0.1019
wide lips	0.0945
bleeding lips	0.0897

(b) Top-5 attributes by the similarity score S_{text} only.

Attribute a_i	S_{sim}	S_{uni}
lipstick	0.1270	1.0000
lip fillers	0.0808	0.1411
red hair	0.0696	0.2173
yellow teeth	0.0687	0.1870
surprised	0.0530	0.2036

(c) Top-5 attributes weighted by the uniqueness score S_{uni}.

Table 2.3: **Validation of our analysis backbone.** The list (a) is added to attribute candidates in an attempt to replace lipstick as the top attribute. (b) and (c) show the analysis results without and with uniqueness score. This table demonstrates the validity and robustness of our counterfactual analysis module.

attributes with text labels. Given a counterfactual pair with semantic perturbations applied (*e.g.*, "lipstick"), we add new candidates that are ambiguous with the differing attributes (*e.g.*, "lip gloss"). Then we evaluate the suitability of CLIP by whether the original results are still ranked first in the counterfactual analysis, over the injected distractor set.

To provide a visual example, we selected two counterfactual pairs from our previous experiment artificially biasing eyeglasses classifier on "lipstick", as depicted in Fig. 2.9. In these comparisons, the most prominent distinction was the presence of lipstick in both pairs. To assess the reliability and robustness of the CLIP model, which serves as our analytical foundation, we introduced an additional set of potentially ambiguous attribute candidates, detailed in Tab. 2.3(a). These attributes shared similarities with lipstick but were contextually incorrect in the given image pairs. Tab. 2.3(c) presents the top-5 matched attributes, with lipstick ranking first by a significant margin in terms of the similarity score S_{sim}. The remaining four attributes were also valid matches, as corroborated by their presence in Fig. 2.9. This showcases the effectiveness of our uniqueness score during the analysis phase. In contrast, Tab. 2.3(b) displays the top-5 matched attributes without the uniqueness mechanism, where the analysis is dominated by lip-related attributes, leading to potential redundancy. The uniqueness score not only highlights distinct candidates but also helps mitigate the inclusion of incorrect attributes, such as "bleeding lips", in the top results.

2.5 Conclusion and Future Work

To the best of our knowledge, UMO presents the first unsupervised approach for diagnosing computer vision models using counterfactual examples. Our method involves optimizing edit vectors within the generative latent space and subsequently analyzing their semantic implications through foundation toolkits. When applied to a target model, our pipeline UMO can autonomously generate a comprehensive diagnosis. This diagnosis includes both visual counterfactual explanations and textual descriptions of vulnerable attributes, all achieved without any human intervention.

We demonstrate the efficacy of our method across a range of vision tasks, encompassing classification, segmentation, and keypoint detection. Through extensive experimentation, we illustrate how UMO excels in producing high-quality counterfactual examples and effectively identifies semantic biases, offering a quantitative assessment of the target model. By conducting cross-model consistency evaluations and incorporating counterfactual training, we establish UMO as a versatile approach for discovering biases and enhancing model robustness.

In this thesis, we have operated under the assumption that the integrated foundation toolkits possess the requisite capability for our diagnostic task. Additionally, our

observations suggest that the DDPM edits encounter challenges due to the limitations of the Asyrp latent space, which lacks full expressiveness. For future directions, we aspire to explore more expressive and disentangled latent spaces within generative models, aiming to enhance the efficiency of counterfactual optimization.

Chapter 3

Model Improvement

3.1 Introduction

The swift progression of computer vision in the past decade can be attributed to improved deep learning algorithms for large-scale training, increased computing power, and the availability of vast datasets such as ImageNet [22] and LAION-5B [108]. While such internet-scale real-world datasets allow to train general vision models, they are not tailored to application scenarios with specific data distributions, *i.e.*, the **cross-distribution** adaptation, which can lead to serious concerns in reliability [63, 104]. This issue often requires costly data collection where models operate.

Among various solutions to this issue, data augmentation has been explored to alleviate such extensive data collection. However, images generated with traditional data augmentation through flipping, gamma adjustments, noise, or more sophisticated methods [20, 137] often fail to align the augmented data with shifted test distributions. Although there are cross-domain augmentation techniques [72, 71], these strategies are task-specific and not easily transferable to other problems. Besides these data-centric efforts, unsupervised domain adaptation (UDA) is an active field of research for such problems from the model aspect (*e.g.*, [46]). Our approach distinguishes itself from the above by its ability to produce *endless* data with much more variability and the need for much fewer samples (*i.e.*, **few-shot**).

To mitigate the distribution discrepancy issues, synthetic datasets have also been studied as a more controllable, diverse, high-quality supplement to the training dataset. Traditionally, simulators and graphics engines are the primary sources of synthetic datasets [126, 115, 91]. However, they typically suffer from unrealism (*i.e.*, domain

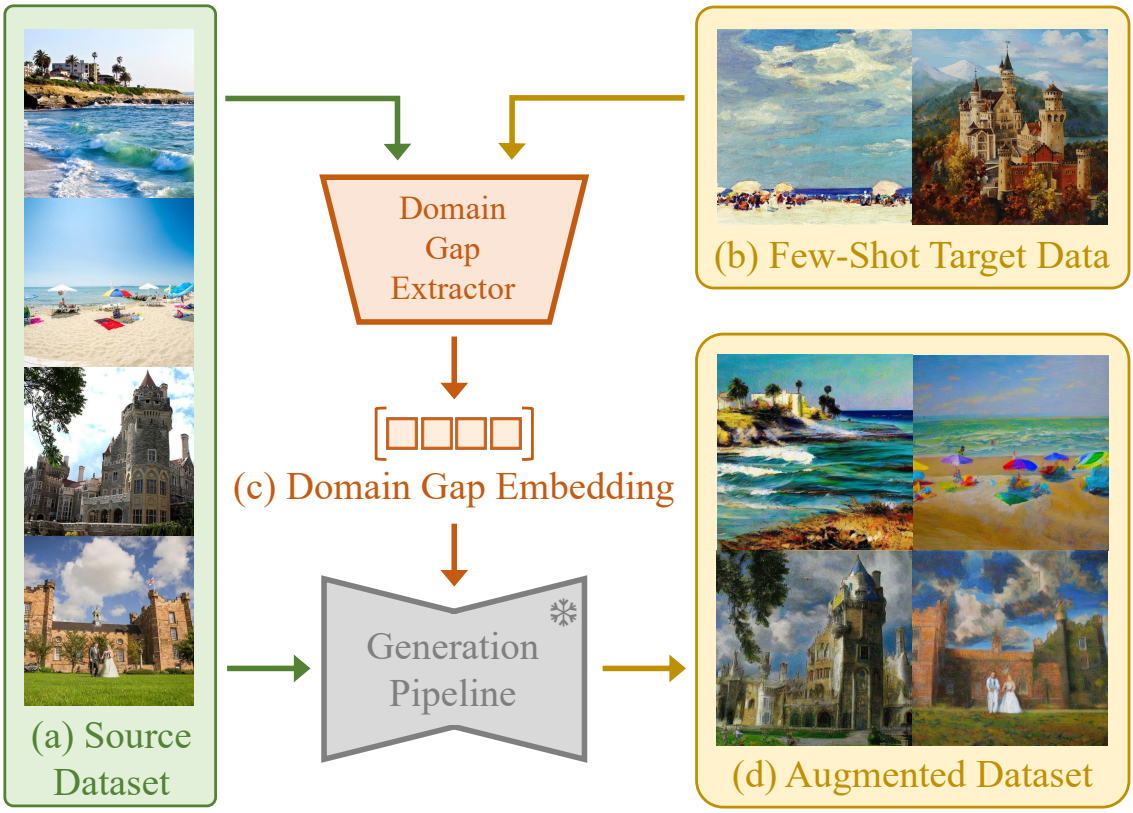

Figure 3.1: **Overview**. In real-world applications, computer vision models often suffer from discrepancies between training and testing data distributions. To alleviate this problem, we propose a novel dataset augmentation method to complement the training dataset with synthetic images. Given (a) a source dataset (*e.g.*, real photos), and (b) a few samples from a target distribution (*e.g.*, paintings), we extract the distribution differences into (c) Domain Gap Embeddings, which enables generating (d) augmented synthetic data to enhance the model performance.

gap) and bounded diversity [38]. With the advancement of visual generative models, they are leveraged for in-domain dataset synthesis in recent works [141, 50, 2]. Nonetheless, very few dataset generation methods [5] focus on the cross-distribution setting guided by just a few input target samples (*e.g.*, 20 images), which is realistic in many scenarios of interest. Moreover, to the best of our knowledge, none achieves target dataset synthesis in such a setting without fine-tuning. The question that we try to address in this thesis is: *Can we use off-the-shelf large pre-trained models (LPMs) as synthetic data generators for effective few-shot dataset augmentation*

towards specific data distributions?

To address this question, we propose DoGE, a few-shot cross-distribution dataset generation framework that is task-agnostic and **inference-only**, as shown in Figure 3.1. The framework takes (a) a source distribution (*i.e.*, the original training dataset), and (b) a few samples from a target distribution in the application context. We propose to extract the distribution discrepancies (*e.g.*, semantic changes, style transfer) into (c) representations in the CLIP latent space [96], named the Domain Gap Embeddings. We then utilize the extracted gap representations to augment source data to generate (d) synthetic datasets that follow the same distribution as the provided few target images.

Our method successfully generates synthetic supplementary datasets as long as (1) the latent representation space, CLIP, has the capacity to express the distribution differences, and (2) the generative diffusion models, Stable UnCLIP [102], is capable of generating in the target distribution. Under these loose constraints, we show that our synthetic datasets from DoGE significantly improve model performance in various computer vision tasks, including subpopulation shifts and domain adaptation. Moreover, DoGE is compatible with and complementary to parallel methods such as UDA and fine-tuning. In summary, DoGE provides the following contributions:

- **Accessibility**: Our framework offers a plug-and-play dataset augmentation experience. With a source dataset to augment, users only need to provide *a few unlabeled images from the target distribution* to obtain an effective synthetic dataset in the desired domain.
- **Efficiency**: Our cross-distribution dataset augmentation framework generates data in the target domain *without the need for fine-tuning*. We directly take advantage of public LPMs, and each step can be inference-only.
- **Effectiveness**: The synthetic datasets from our generation pipeline can successfully improve the task model performance by a significant margin.

3.2 Related Works

While real-world images are cornerstones of computer vision, as modern vision datasets increase in size, it has become gradually more challenging to scrutinize and clean the collected data. The difficulty of curating large datasets poses potential issues such as noisy labels and dataset imbalance [12, 7, 84]. Hence synthetic data became a popular alternative with high controllability and accessiblility.

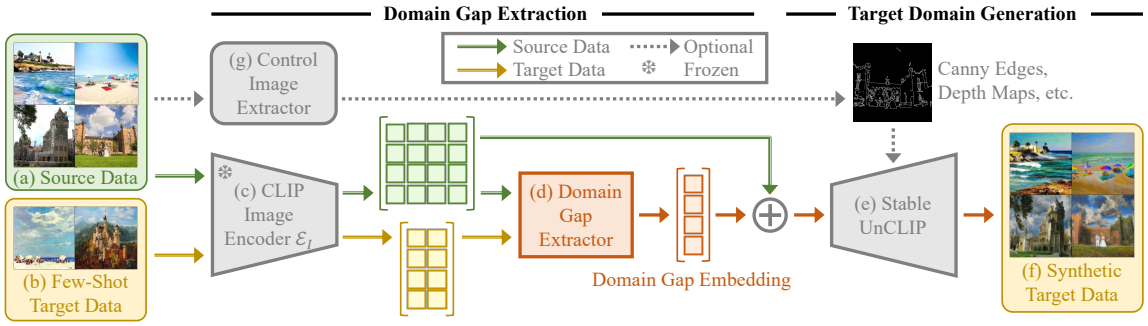

Figure 3.2: **Framework**: (a) The source dataset and (b) a few target data samples are first (c) encoded in the CLIP embedding space. We then (d) extract the representation, named the Domain Gap Embedding, between the source and target distributions. The Domain Gap Embedding augments source image embeddings to construct the latent input to (e) the generative model (Stable UnCLIP), which generates (f) a synthetic dataset following the target distribution. Optionally (dotted lines), we can (g) integrate ControlNet to provide further structural guidance to preserve the source image structures.

Generative Models for Image Data: Recent advances in generative models have provided powerful tools for synthetic data generation. Generative Adversarial Network (GAN) pioneered a new direction for high-quality image synthesis [31, 8, 59, 56]. In parallel, diffusion models [42, 83, 114, 43] demonstrate their promising potential, leading to many astonishing works including GLIDE [82], DALL•E 2 [98], Imagen [105], and Stable Diffusion [102].

Besides generative backbones, fine-grained controllability of the generative models is also essential for data synthesis. In the direction of GANs, CycleGAN [144], CyCADA [44], and CLIP-enabled methods [89, 145] achieved effective image-to-image transfer and targeted editing toward desired distributions. For diffusion models, various conditioning techniques regulate the generations. Some methods [40, 9] leverage the cross-attention maps to apply accurate prompt-based augmentations. Other works [28, 121, 35, 136] learn special tokens and embeddings to preserve identities during data generation. Similarly, methods in [47, 103, 36, 60, 62] fine-tune the diffusion models for desired generation, while image-to-image synthesis [127, 79] is also critical to data augmentation. Finally, ControlNet [139] uses condition maps to control the generation accurately.

Synthetic Data Generation: With such extensive generation capability and fine-grained controllability, generative models have been leveraged to populate synthetic datasets [50, 13]. GANs have been used for effective synthetic dataset generation through latent space manipulation [6, 141, 67, 76]. Enabled by the abundant generation controls in diffusion-based networks, more recent works leverage diffusion models to improve data diversity by expanding existing datasets [112, 4, 94, 142, 120, 117].

While the above methods can generally expand a given dataset, they suffer from subpopulation and domain shifts in datasets. Regarding subpopulation shifts, Fill-Up [111] incorporates Textual Inversion [28] to fix imbalanced datasets by uneven generation but requires optimizing a token for each class. To address domain shifts, some methods [133, 24] utilize captioning models to describe target distributions and construct new prompts for generation. However, since the expressibility of texts is limited, other methods also resort to fine-tuning for adaptation. Assuming access to the full target dataset, solutions in [2, 90] fine-tuned Imagen and DDPM [42] for better in-domain generations. Under the few-shot setup where only a few target samples are available, DomainStudio [143] introduced similarity loss to conquer the over-fitting issue in fine-tuning, and DATUM [5] proposed to fine-tune the model into the target domain with crops of the few target samples. Nonetheless, such methods require domain-specific fine-tuning and may introduce training algorithm modification, while our method, with better performance in our experiment setups, can be directly applied off the shelf for the given target images.

3.3 Method

Recognizing the lack of practical and readily available cross-distribution dataset synthesis methods, we introduce a novel, model-agnostic few-shot dataset augmentation framework. Our framework possesses the ability to create synthetic samples that conform to the target distribution based on a minimal set of input images. It is characterized by its simplicity and effectiveness, and, in its fundamental configuration, does not necessitate any training.

Our framework consists of two main components: modeling the domain gap and generating across the domain gap, shown in Fig. 3.2. To generate from one dataset distribution to another, we first capture the differences between them as Domain Gap Embeddings, shown in Sec. 3.3.1. With the representation for the distribution gap, Sec. 3.3.2 illustrates our method for generating datasets from the source to target

distribution, with an optional trick to preserve image quality. To further improve the usefulness of the generated dataset, we also conduct confidence-based generation cleaning methods on downstream tasks, shown in Sec. 3.3.3.

3.3.1 Domain Gap Extraction

When capturing differences in data distributions, fine-tuning generative models across domains can be costly, and prompts may not articulate the discrepancies. Hence, we focus on modeling the distribution differences in the latent space. The recent research in visual representation learning introduces powerful semantic latent spaces such as CLIP. CLIP is assumed to have sufficient knowledge generalization for common settings, and its linear vector compositionality enables semantically meaningful operations [118]. In our framework, we choose to leverage the CLIP latent space to capture the gap between data distributions and directly apply it in data augmentation. Such captured distribution discrepancies are named the Domain Gap Embeddings.

Fig. 3.2 (left) shows the domain gap extraction process. The input consists of a source dataset $\mathcal{D}_\mathcal{S}$ (Fig. 3.2a), with $|\mathcal{D}_\mathcal{S}| = N$, and a few data samples $\mathcal{D}_\mathcal{T} = \{y_j\}_{j=1}^m$ (Fig. 3.2b) from a different target distribution with $m \ll N$. We first encode images from a randomly sampled subset $\hat{\mathcal{D}}_S = \{x_i\}_{i=1}^n \subseteq \mathcal{D}_\mathcal{S}$ and $\mathcal{D}_\mathcal{T}$ into the CLIP space via a CLIP image encoder $\mathcal{E}_\mathcal{I}$ (Fig. 3.2c). Denoting the image embeddings as $z_{x_i} = \mathcal{E}_\mathcal{I}(x_i)$ and $z_{y_j} = \mathcal{E}_\mathcal{I}(y_j)$, we study two options as the Domain Gap Extractor (Fig. 3.2d) to capture the gap representation Δz. A straightforward way is computing the expected differences of all pairs between the source and target dataset, which is equivalent to the difference of the means of the images assuming $\mathcal{D}_\mathcal{S}$ is independent of $\mathcal{D}_\mathcal{T}$, *i.e.*,

$$\Delta z = \mathbb{E}_{x_i \in \mathcal{D}_\mathcal{S}}[\mathbb{E}_{y_j \in \mathcal{D}_\mathcal{T}}[\mathcal{E}_\mathcal{I}(y_j) - \mathcal{E}_\mathcal{I}(x_i)]] \tag{3.1}$$

$$= \frac{\sum_{j=1}^m z_{y_j}}{m} - \frac{\sum_{i=1}^n z_{x_i}}{n}. \tag{3.2}$$

Another way to extract the gap is through Principal Component Analysis (PCA) [27]. Since the first principal direction from PCA denotes the direction where a distribution varies the most, we leverage this property and apply PCA on a joint set $\{z_{x_i}\}_{i=1}^n + \{z_{y_j}\}_{j=1}^m$ with $n = m$. The first principal direction from PCA is then considered as the domain gap representation Δz. From empirical results shown in Appendix B.1, we observe that the first option of computing the domain gap (Eq. (3.2)) yields better generation quality. The impact of the values of n and m is also addressed in

Appendix B.1. For the rest of this work, we adopt this mechanism as our Domain Gap Extractor, yet users can easily design and swap in their own extractor in our framework.

3.3.2 Target Dataset Generation

With the domain gap extracted into the latent form Δz, we augment source images to generate the synthetic dataset, as shown in the right half of Fig. 3.2. Capturing the gap Δz between distributions in the CLIP space opens up methods to generate data across domains. While various diffusion-based generative models accept texts or images as input or conditions, our method contrasts with these in that we directly interact with CLIP latent embeddings. Such is made possible by the UnCLIP approach introduced in DALL•E 2, specifically the Stable UnCLIP model [102] (Fig. 3.2e). It is a fine-tuned Stable Diffusion model that accepts CLIP image embeddings directly as input. Hence, in this work, we choose the Stable UnCLIP model, denoted as G as our generation backbone in the framework.

Given the distribution gap Δz and source image embeddings $\{z_{x_i}\}$ both in the CLIP space, we augment the source image representations by simply adding the gap Δz to them. To further increase diversity, we also introduce small Gaussian random perturbations $\epsilon \sim \mathcal{N}(\mathbf{0}, 10^{-3}I)$ in the augmentation. Similarly, we introduce a distributional edit strength scalar $C \sim \mathcal{N}(c, 0.05)$. The impact of values of c is discussed in Sec. 3.4.4. Hence, the generated k images, denoted as $\{\hat{y}_i\}_{i=1}^{k}$ (Fig. 3.2f) are obtained as:

$$\hat{y}_i = G(z_{x_i} + C \cdot \Delta z + \epsilon). \tag{3.3}$$

The above two steps form our base framework and can already achieve effective target dataset generation (details in Sec. 3.4). Nonetheless, in specific cases, additional techniques can be adopted for higher generation quality.

Finer Generation Control: In some cases, it is beneficial to preserve the visual structure of the original source data to be augmented. For example, maintaining the object structure can further ensure less deformation or corruption in the generation. Because the expressiveness of one vector in the CLIP space is limited, a fine-grained structural control can introduce more detailed visual guidance on top of our domain gap embeddings.

Therefore, we integrate ControlNet into our generative module for accurate image

structure control during the generation. As shown in Fig. 3.2g, we can feed the input source data through a control image extractor. Given a source image, this module outputs a series of domain-invariant control maps for generation, including Canny edge maps [14] and HED edge maps [131], and processing depth and segmentation maps from ground truth labels of source data, if available. During the generation phase, we feed these control maps into our revised Stable UnCLIP model for more refined generations. This guidance enriches our augmentation with the compositional information, which empirically brings further improvements as shown in Sec. 3.4.3.

3.3.3 Confidence-Based Generation Cleaning

While the ControlNet integration preserves the structure and quality of our generated datasets, it is not safe to assume that every synthetic image is valid and helpful data. Inspired by [85], we propose a confidence-based filtering mechanism to remove such poor generations. Given a downstream task model trained on the source data, *e.g.*, a classifier, we fine-tune this model with our data augmentation to improve test performance. At each iteration during fine-tuning, before training we first perform inference with the current model to filter out augmented data with highly confident but incorrect predictions. The confidence is determined as the highest predicted softmax score s among all classes. With a threshold parameter t, we discard synthetic samples where the model prediction is wrong but its confidence is greater than the threshold, *i.e.*, $s > t$. We only temporarily discard samples in each training step but never eliminate any data from the dataset. Please see Appendix B.2 for details.

3.4 Experiments

This section illustrates the versatility and efficacy of DoGE, demonstrating its ability to produce synthetic datasets benefiting various computer vision challenges. Sec. 3.4.1 presents the standard experimental setups employed in our studies. Subsequently, Sec. 3.4.2 addresses issues related to imbalanced class distributions and showcases effectiveness under the presence of spurious correlations. In Sec. 3.4.3, we delve into the effectiveness of our dataset generation approach under common domain adaptation problems. In addition to task-based evaluations, we conduct ablation studies concerning our generative pipeline in Sec. 3.4.4. These studies serve to provide both qualitative and quantitative assessments of the synthetic datasets created through

DoGE.

3.4.1 Experimental Setup

Baselines: For classification tasks in Sec. 3.4.2 and Sec. 3.4.3, *base* refers to the models trained on the source data in the cross-domain setting only. Subsequently, we fine-tuned the *base* models on the augmented datasets to assess the efficacy of data generation methods. We compared against one traditional augmentation, RandAugment [20], and two generative methods, DA-Fusion [117] and DATUM [5]. For fair comparisons, we kept the number of generated images the same within each task across all generative methods.

Implementation: For *base* classification models we fine-tuned ImageNet pre-trained ResNet50 [37] models for 20 epochs with AdamW optimizer [74] at a constant learning rate of 10^{-3} and a batch size of 128. For each generative baseline and our method, the *base* model is further fine-tuned on respective augmented dataset for 20 epochs, with AdamW optimizer and a batch size of 256. For CelebA, we used a constant learning rate of 10^{-3}. For DomainNet and FMoW, the classification head was trained with a learning rate of 10^{-4}, and its preceding layers with 10^{-5}. For all datasets, confidence-based generation cleaning was applied at training time with a threshold $t = 0.9$. After fine-tuning, the final models were saved for evaluation. We also extended to segmentation problems where we directly adopted the synthetic data evaluation pipeline generously published in DATUM, the current state-of-the-art method in one-shot UDA for self-driving segmentation problems.

3.4.2 Subpopulation Shift

In our initial experiment, we sought to assess the effectiveness of our solution in addressing the subpopulation shift problem. Specifically, we aimed to evaluate how well DoGE could mitigate imbalanced training data distributions that result in spurious correlations.

We curated subsets of facial data from the CelebA dataset, intentionally introducing imbalances in certain attributes. Given an attribute (*e.g.*, perceived gender), we selected a secondary attribute (*e.g.*, eyeglasses), as the bias factor. Then, we sampled 1000 males wearing eyeglasses and 1000 females without eyeglasses, denoted as the source (majority distribution) in Fig. 3.3a. We also sampled 20 images per class

Figure 3.3: **Examples of synthetic CelebA data generated from (a) Source into (b) Target distribution.** Under subpopulation shift, we generated data from the majority subpopulation (a) into the under-represented distribution (b). (c) shows the synthetic data generated from our pipeline. The results demonstrate our capability to apply semantic augmentation in accordance with gaps between distributions.

Method	Test Accuracy (%)
Base	38.00
Oversampling	53.80
RandAugment [20]	62.40
DA-Fusion [117]	59.72
DoGE (Ours)	**67.16**

Table 3.1: **Test Accuracy on our constructed CelebA imbalanced classification problem.** We evaluated our method against four baselines. This table shows that synthetic data from DoGE has a significant advantage over other methods.

Method	Test Accuracy (%)
Base	38.00
LoRA [47]	56.05
LoRA + DoGE (Ours)	**74.28**

Table 3.2: **Test Accuracy on CelebA imbalanced classification problem with fine-tuned generative models.** We applied our method on top of a personalized generator via LoRA and show that DoGE is complementary to adaptation via personalization.

with the opposite secondary attribute (*i.e.*, bias), denoted as the target (minority distribution) in Fig. 3.3b. Training a perceived gender classification model on this imbalanced subset naturally introduced bias toward eyeglasses over gender.

To supplement this imbalanced training set, we first extracted the distribution

Method	Test Acc (%)	Δ
Base	32.86	—
DoGE	38.64	+5.78
DoGE + ControlNet	40.29	+1.65
DoGE + ControlNet + Cleaning	41.30	+1.01

Table 3.3: **Incremental improvements on DomainNet (Real → Painting) problem.** We gradually added our components to the base model and evaluated the effectiveness of each part.

gap for each class from randomly sampled only 10 source and 10 target images. Then, DoGE generated 1000 synthetic samples per class in the target (minority) distributions with $c = 1.0$. Fig. 3.3c shows the generated images following the target distribution where eyeglasses are successfully added or removed respectively to follow the under-represented subpopulation.

After the data generation, our new training set consists of 1000 sampled source data (Fig. 3.3a) and 1000 generated target data (Fig. 3.3c). For the test set, we sampled 1000 images per class from the target (minority) distribution (Fig. 3.3b) in CelebA. For comparison, we first oversampled target data by duplication, then applied RandAugment to this oversampled training set. DA-Fusion was also used to expand each class in the target data. All baselines generated the same amount of data in the evaluation as ours. Tab. 3.1 shows the test accuracy after training on our synthetic data along with the baseline performances to compare with. DoGE achieved the best test accuracy among the baselines.

Since fine-tuning is studied as a powerful method for targeted generation, we demonstrated our compatibility with LoRA [47] and generated synthetic data using a fine-tuned Stable UnCLIP model. The results in Tab. 3.2 indicate that our method complements adaptation via fine-tuning.

3.4.3 Unsupervised Domain Adaptation

Classification Tasks

DomainNet consists of 0.6 million images of 345 classes distributed across 6 unique domains including Real (**R**), Clipart (**C**), Infograph (**I**), Painting (**P**), Quickdraw (**Q**) and Sketch (**S**). We evaluated our method on 4 domain adaptation tasks from **R** to **P**, **S**, **C** and **I**, using official test sets. In each task, we randomly sampled 345

Figure 3.4: **Examples of synthetic DomainNet data, generated from source data into four different target domains.** Each generation (bottom) was augmented from the source image (top) using our pipeline with ControlNet. The results demonstrate our capability to augment data in accordance with gaps between distributions.

images from both the source (*i.e.*, **R**) and target (*i.e.*, **P**, **S**, **C** or **I**) distribution to calculate the domain gap. Synthetic **P** and **S** images were generated with edit strength mean $c = 1.3$, **I** with $c = 1.1$, and **C** with $c = 1.5$. Sec. 3.4.4 discusses the choice of values for c. For each class DoGE generated 128 images with ControlNet (Fig. 3.4) to supplement the training data for fine-tuning, increasing the dataset size by approximately 30%.

Tab. 3.3 shows the incremental improvements of training on Real domain and testing on Painting domain. Stand-alone DoGE improved w.r.t standard approaches and additional techniques further increased our advantages. Tab. 3.4 shows full comparisons on all four domains, and DoGE achieved the best accuracy in all settings.

To demonstrate DoGE's compatibility and improvements to traditional UDA solutions, we evaluated our performance based on six UDA methods. For each method, we incorporated DoGE by simply adding our synthetic images to the training dataset. Tab. 3.5 shows that our method can help further improve UDA methods in general. Please see Appendix B.4 for the complete experiment.

FMoW-WILDS [63] is a modified version of the Functional Map of the World [18] dataset. It includes 0.5 million RGB satellite images labeled with 62 land use categories, with domains defined by their captured years and geographic regions spanning Asia, Africa, Americas, Oceania and Europe. For this thesis, we focused on the domain adaptation performance across different time periods within three regions: Asia,

Method	DomainNet Acc (%)				FMoW Acc (%)		
	Painting	Infograph	Clipart	Sketch	Asia	Americas	Oceania
Base	34.64	14.48	39.06	24.70	66.27	64.65	74.42
RandAugment [20]	37.20	15.90	41.08	26.26	64.35	70.47	74.29
DA-Fusion [117]	39.57	16.54	42.22	28.27	70.97	76.71	77.32
DATUM [5]	38.19	17.80	40.96	29.46	71.63	78.35	75.24
DoGE (Ours)	**44.00**$^{\pm.0}$	**18.71**$^{\pm.0}$	**45.61**$^{\pm.3}$	**34.96**$^{\pm.3}$	**72.62**$^{\pm.1}$	**78.94**$^{\pm.2}$	**78.14**$^{\pm.1}$

Table 3.4: **Test accuracy in unsupervised domain adaptation classification problems.** We evaluated against four baselines on the left column. For DomainNet, the task is to adopt a model with a Real domain training dataset to Painting, Infograph, Clipart, and Sketch domains. For FMoW, for each region (Asia, Americas, Oceania), we adopted a model with old satellite images (2002-12) to perform well on new satellite data (2016-17). The table shows that our methods achieved the highest test accuracy in every category.

UDA Method	Test Acc (%)		
	w/o DoGE	w/ DoGE	Δ
BSP [16]	46.76	47.34	+0.58
DANN [29]	47.01	49.68	+2.67
CDAN [73]	51.66	52.11	+0.45
MCD [106]	50.88	52.14	+1.26
MCC [53]	50.08	52.95	+2.87
MemSAC [55]	52.27	54.16	+1.89

Table 3.5: **Test Accuracy of UDA methods on the DomainNet (Real → Painting) problem.** We evaluated existing UDA methods with and without DoGE. The table shows that our approach is compatible with and complementary to UDA methods.

Americas, and Oceania. Specifically, within each region, the source and target domain refer to the satellite images taken between 2002-12 and 2016-17. We randomly sampled 64 images from each domain to calculate the gap. For each land use category, DoGE generated 64 images using ControlNet (*e.g.*, Fig. 3.5) with edit strength mean $c = 1.3$, accounting for approximately 10% increase in the dataset size. Tab. 3.4 shows that DoGE leads to higher performance than baselines in all experiments.

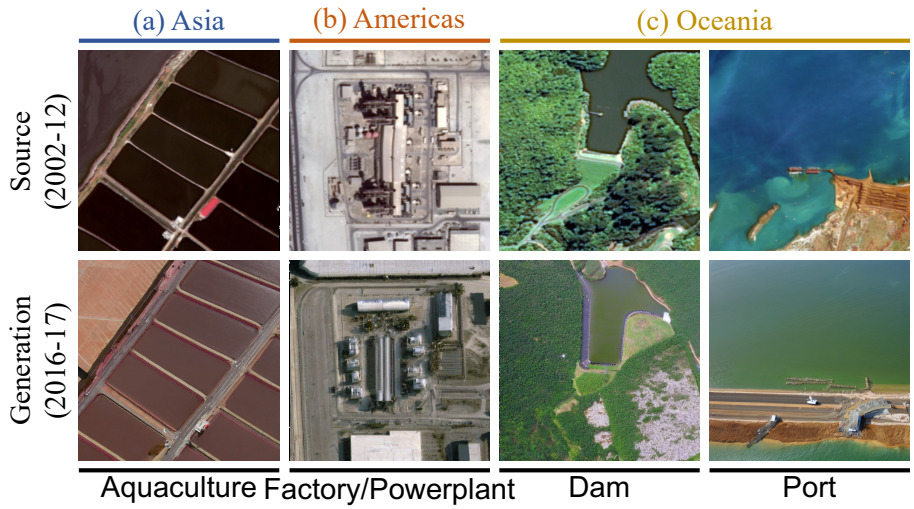

Figure 3.5: **Examples of synthetic FMoW data**, generated from Source (2002-12) into Target (2016-17) distributions in 3 regions using our method with ControlNet. The results illustrate our capacity to generate images across temporal discrepancies.

Segmentation Task

Besides classification problems, our method is also generally applicable to other computer vision tasks. To illustrate the versatility and generality of DoGE, we demonstrate our capability to improve cross-domain segmentation problems.

In this experiment, we chose GTA5 [99] as our source domain and Cityscapes [19] as our target domain. Using the full GTA5 dataset and 20 unlabeled images from the Cityscapes, we generated synthetic data in Fig. 3.6 with and without ControlNet. We evaluated our synthetic generation under the scope of UDA. As baselines, we chose DAFormer [45], a UDA segmentation method, and DATUM combined with DAFormer. Similar to DATUM, we evaluated our method on top of DAFormer, *i.e.*, expanding the unlabeled data available to DAFormer. Tab. 3.6 shows DoGE is able to achieve at-par performance with DATUM. Moreover, DATUM requires fine-tuning a Stable Diffusion model while ours is inference-only in a plug-and-play fashion.

Figure 3.6: **Examples of synthetic self-driving data generated from (a) GTA5 source images into (b) Cityscapes target domain.** (c) shows the synthetic data generated from our pipeline without any improvement tricks. We also demonstrated the generation with scene structure preserved by ControlNet (conditioned on canny edges and source segmentation ground truth) in (d). The synthetic data are then used in unsupervised domain adaptation methods to adapt models across domains.

Method	Test Accuracy (%)
DAFormer [45]	48.2
DAFormer + DATUM [5]	56.4
DAFormer + DoGE (Ours)	**57.3**

Table 3.6: **GTA5 → Cityscapes cross-domain segmentation.** We used DAFormer as our UDA baseline. DATUM and DoGE are target data generators applied on top of DAFormer. Our performance is at par with DATUM while exempt from any training.

3.4.4 Ablation Studies

Generation Quality

The usefulness of our synthetic data is directly dependent on the generation quality. We focus on two aspects to assess the generated images: the FID score [41] for image quality, and the t-SNE [119] for distribution alignment.

The exploration was conducted under our DomainNet Real→Painting experiments. Tab. 3.7 shows that under FID metrics with respect to DomainNet Painting images, our generation achieved the best quality with respect to the Painting data from DomainNet. To visualize the distribution alignment of our generation, we plotted the t-SNE graph of source (Real domain), target (Painting domain), and our synthetic painting images in Fig. 3.7. It shows that our synthetic data are successfully augmented into the target distribution and away from the source distribution.

Data Source	FID Score (↓)
Source Data	30.98
DA-Fusion [117]	40.20
DATUM [5]	219.00
DoGE (Ours)	24.86
DoGE w/ ControlNet (Ours)	**18.25**

Table 3.7: **FID scores against the DomainNet painting images.** We evaluated the FID scores against the DomainNet Painting samples on the DomainNet Real images, the synthetic data from [117] and [5], and our generations. The table shows that our synthesis achieved the best FID score among the baselines.

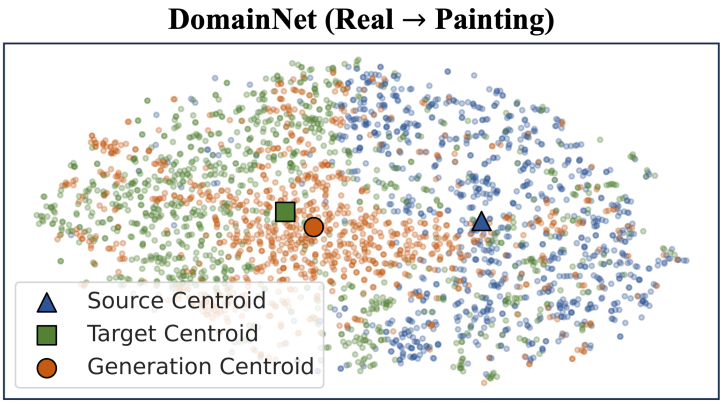

Figure 3.7: **The t-SNE plots of the source, target, and generated data.** In our DomainNet Real → Painting experiment, we drew a t-SNE plot to visualize distributions of source, target, and our generation. Our generation is well-aligned with the target distribution.

Domain Gap Embedding Editing Weights

One of the important hyper-parameters that impacts the generation is the edit strength scalar C defined in Sec. 3.3.2. In this section, we study the effect of different deterministic values for C visually to better understand the domain gap embeddings. To isolate the effect, we do not apply ControlNet in this experiment. As shown in Fig. 3.8, we conduct the exploration in two settings: face augmentation with eyeglasses as the distribution gap (top row), and object augmentation from the real domain to the sketch domain (bottom row). Starting with the source image (left-most column), we

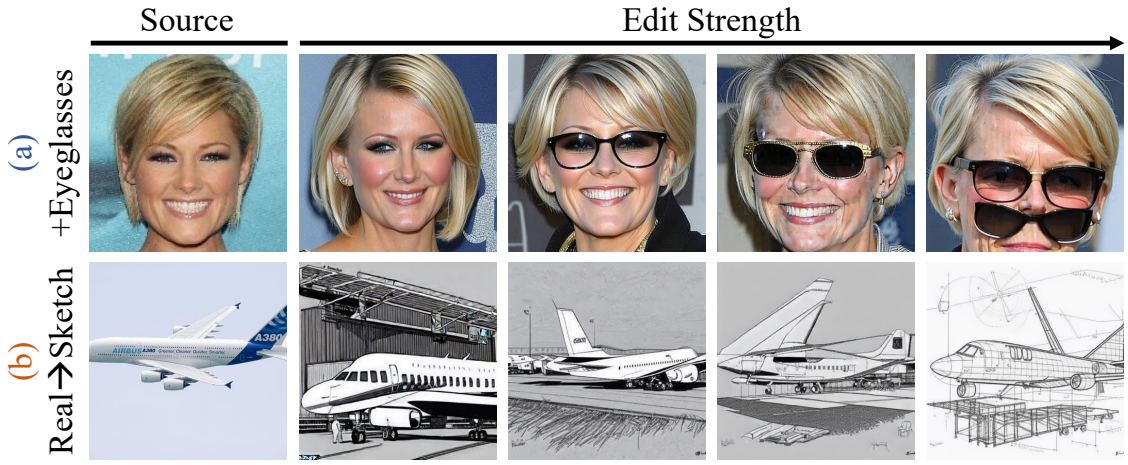

Figure 3.8: **Effect of increasing edit strength** c**.** We considered two source images under two tasks: (a) adding eyeglasses to faces and (b) converting real to sketch images. In each task, we generated images with gradually increasing edit strength. At the right end, we observe that two glasses are added $c = 2.0$ and the most sketchy airplane $c = 2.5$. As expected, the edit strength dictates the extent of emphasis on the distribution differences.

gradually increase the edit strength C and generate images at each different value for C. For the face, we set C to 0.5, 1.0, 1.5, 2.0 from left to right, and 1.0, 1.5, 2.0, 2.5 respectively for the airplane.

From Fig. 3.8, we observe that the magnitude of the edit strength impacts the extent of our augmentation. When $C \geq 2$ as shown in the right-most column, our pipeline adds two glasses on the face indicating an over augmentation. Meanwhile, for the airplane, the value of C influences whether the generation is a realistic sketch or a simple sketch. Hence the best choice of edit strength mean c should be assessed based on the kind of task in practice.

3.5 Conclusion and Future Works

This thesis introduces DoGE, an innovative diffusion-based data augmentation technique designed to address cross-distribution challenges. Our method is distinguished by its accessibility, efficiency, and remarkable effectiveness. We utilize Domain Gap Embeddings, which capture distribution differences, as direct augmentations applied to source data embeddings. Our generative backbone, Stable UnCLIP, is leveraged

to facilitate this process. It's worth noting that our pipeline operates without the necessity for training, relying exclusively on a minimal set of images from the target distribution to guide the augmentation process. The result is the generation of diverse and high-quality synthetic data, which significantly enhances test performance.

We showcase the versatility and effectiveness of our method across various problem settings. Notably, our approach not only excels at transferring styles but also introduces semantic augmentations according to distribution disparities. In comparison to other general data synthesis methods, we achieve the highest improvements across all tasks. We also highlight the adaptability of DoGE by evaluating its performance in a segmentation task, demonstrating competitive inference-only results compared to the state-of-the-art method, which requires training. Furthermore, we illustrate that our approach is compatible with and complementary to parallel strategies such as UDA and fine-tuning.

While our method boasts significant strengths, it does have certain limitations that merit further consideration. First and foremost, the expressiveness of the CLIP model's latent space can be constrained when confronted with domain gaps that CLIP is unfamiliar with. Additionally, while Stable UnCLIP proves effective in numerous real-world scenarios, it may face challenges in out-of-domain situations, such as medical X-ray imagery. Lastly, there is ample room for exploration in devising more effective training algorithms to maximize the utility of synthetic data.

Chapter 4

Conclusion

At the dawn of a new era in artificial intelligence, the evolution of foundation models brings a promising horizon for enhancing AI trustworthiness. This thesis, through its exploration of diagnosing and improving computer vision models using generative models and foundation models, contributes significantly to this booming field. It encapsulates a journey from identifying the inherent vulnerabilities and biases present in current vision systems to rectifying these through targeted synthetic data augmentation, thereby contributing towards more reliable, fair, and robust AI systems.

The integration of Unsupervised Model Diagnosis (UMO) and Domain Gap Embeddings (DoGE) presents a cohesive, innovative approach that not only identifies but also addresses the nuanced challenges faced by computer vision models. UMO's capability to diagnose models' vulnerabilities without requiring annotated datasets or human intervention paves the way for automated understanding of the models' issues. Concurrently, DoGE's use of synthetic data to bridge domain gaps and enhance model performance exemplifies the benefits of generative models in crafting dataset solutions that are not only effective but also scalable and adaptable to various application domains. Together, UMO and DoGE offer a comprehensive two-phase pipeline that automatically diagnoses and treats model vulnerabilities without the need for extensive human supervision, thereby advancing and democratizing the fairness, trustworthiness and robustness of AI deployments.

Moreover, this work paves the path forward as foundation models continue to evolve. The vast information and knowledge encapsulated in these models open up new promising approaches and directions for enhancing AI trustworthiness. The

methodologies developed in this thesis serve as effective examples for leveraging foundation models not just as tools for task performance but as integral components of a holistic strategy to improve AI's reliability and fairness.

In conclusion, this thesis explores a new direction in utilizing the advancements in generative models and foundation models to enhance the trustworthiness of AI systems. A comprehensive pipeline that democratizes scalable AI model diagnosis and improvement is presented, making one step closer to the trustworthy AI that is widely accessible by anyone. The proposed framework also bridges the efforts between diagnosing model vulnerabilities and effectively addressing them through synthetic data augmentation. As foundation models continue to evolve, the insights and methodologies presented in this work offer valuable perspectives for the AI research community. This thesis not only contributes to the academic frontier but also provides practical frameworks that can be adapted and expanded upon, paving the way for the development of AI systems that are truly trustworthy and aligned with societal values and ethical standards.

Bibliography

[1] Maximilian Augustin, Valentyn Boreiko, Francesco Croce, and Matthias Hein. Diffusion visual counterfactual explanations. *NeurIPS*, 2022. 8, 11

[2] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. In *TMLR*, 2023. 26, 29

[3] Balamurugan Balusamy, Naveen Chilamkurti, Rajesh Kumar Dhanaraj, Rishabha Malviya, and Sonali Sundram. *Artificial Intelligence for Health 4.0: Challenges and applications*. River Publishers, 2023. 1

[4] Hritik Bansal and Aditya Grover. Leaving reality to imagination: Robust classification via generated datasets. *arXiv preprint arXiv:2302.02503*, 2023. 29

[5] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. One-shot unsupervised domain adaptation with personalized diffusion models. In *CVPRW*, 2023. 26, 29, 33, 37, 39, 40, 75

[6] Victor Besnier, Himalaya Jain, Andrei Bursuc, Matthieu Cord, and Patrick Pérez. This dataset does not exist: training models from generated images. In *ICASSP*, 2020. 29

[7] Abeba Birhane and Vinay Uday Prabhu. Large image datasets: A pyrrhic win for computer vision? In *WACV*, 2021. 27

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019. 28

[9] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 28

[10] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1

[11] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 1

[12] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. In *Neural Networks*, 2018. 27

[13] Max F Burg, Florian Wenzel, Dominik Zietlow, Max Horn, Osama Makansi, Francesco Locatello, and Chris Russell. A data augmentation perspective on diffusion models and retrieval. *TMLR*, 2023. 29

[14] John Canny. A computational approach to edge detection. In *TPAMI*, 1986. 32

[15] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, 2019. 2

[16] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *ICML*, 2019. 37, 75

[17] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020. 17, 64

[18] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *CVPR*, 2018. 36

[19] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 38

[20] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPRW*, 2020. 25, 33, 34, 37

[21] A. C. Davison and C.-L. Tsai. Regression model diagnostics. In *International Statistical Review*, 1992. 8

[22] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 18, 25, 64

[23] Ranjie Duan, Xingjun Ma, Yisen Wang, James Bailey, A Kai Qin, and Yun Yang. Adversarial camouflage: Hiding physical-world attacks with natural styles. In *CVPR*, 2020. 5

[24] Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. *arXiv preprint arXiv:2305.16289*, 2023. 29

[25] Sabri Eyuboglu, Maya Varma, Khaled Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Ré. Domino: Discovering Systematic Errors with Cross-Modal Embeddings. In *ICLR*, 2022. 8

[26] W. Nelson Francis and Henry Kucera. Computational analysis of present-day american english. Brown University Press, 1967. 13

[27] Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. 1901. 30

[28] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2023. 28, 29

[29] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015. 37, 75

[30] Rui Gong, Qin Wang, Dengxin Dai, and Luc Van Gool. One-shot domain adaptive and generalizable semantic segmentation with class-aware cross-domain transformers. *arXiv preprint arXiv:2212.07292*, 2022. 78

[31] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 28

[32] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. *ICLR*, 2015. 5, 8

[33] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *ICML*, 2019. 3, 5

[34] Patrick Grother, Mei Ngan, and Kayee Hanaoka. *Face recognition vendor test (fvrt): Part 3, demographic effects*. National Institute of Standards and Technology Gaithersburg, MD, 2019. 2

[35] Inhwa Han, Serin Yang, Taesung Kwon, and Jong Chul Ye. Highly personalized text embedding for image manipulation by stable diffusion. *arXiv preprint arXiv:2303.08767*, 2023. 28

[36] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. In *ICCV*, 2023. 28

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 33, 64

[38] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *ICLR*, 2023. 26

[39] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *CVPR*, 2021. 3

[40] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2023. 28

[41] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. 39, 66

[42] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 7, 12, 28, 29

[43] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPSW*, 2021. 28

[44] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*, 2018. 28

[45] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022. 38, 39

[46] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, 2023. 25

[47] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 28, 34, 35

[48] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. GANSpace: Discovering Interpretable GAN Controls. In *NeurIPS*, 2020. 7

[49] Paul Jacob, Éloi Zablocki, Hedi Ben-Younes, Mickaël Chen, Patrick Pérez, and Matthieu Cord. Steex: steering counterfactual explanations with semantics. In *ECCV*, 2022. 8

[50] Ali Jahanian, Xavier Puig, Yonglong Tian, and Phillip Isola. Generative models as a data source for multiview representation learning. In *ICLR*, 2022. 26, 29

[51] Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Diffusion models for counterfactual explanations. In *ACCV*, 2022. 8, 11

[52] Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Adversarial Counterfactual Visual Explanations. In *CVPR*, 2023. 9

[53] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *ECCV*, 2020. 37, 75

[54] Ameya Joshi, Amitangshu Mukherjee, Soumik Sarkar, and Chinmay Hegde. Semantic Adversarial Attacks: Parametric Transformations That Fool Deep Classifiers. In *ICCV*, 2019. 6, 8

[55] Tarun Kalluri, Astuti Sharma, and Manmohan Chandraker. Memsac: Memory augmented sample consistency for large scale domain adaptation. In *ECCV*, 2022. 37, 75

[56] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *CVPR*, 2023. 28

[57] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training Generative Adversarial Networks with Limited Data. In *NeurIPS*, 2020. 7

[58] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *CVPR*, 2019. 7

[59] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 28

[60] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, 2023. 28

[61] Saeed Khorram and Li Fuxin. Cycle-consistent counterfactuals by latent transformations. In *CVPR*, 2022. 8, 11

[62] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022. 28

[63] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *ICML*, 2021. 25, 36

[64] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *ICLR*, 2023. 7, 11

[65] Oran Lang, Yossi Gandelsman, Michal Yarom, Yoav Wald, Gal Elidan, Avinatan Hassidim, William T. Freeman, Phillip Isola, Amir Globerson, Michal Irani, and Inbar Mosseri. Explaining in Style: Training a GAN To Explain a Classifier in StyleSpace. In *ICCV*, 2021. 8

[66] Bo Li, Qiulin Wang, Jiquan Pei, Yu Yang, and Xiangyang Ji. Which Style Makes Me Attractive? Interpretable Control Discovery and Counterfactual Explanation on StyleGAN. *arXiv preprint arXiv:2201.09689*, 2022. 9

[67] Daiqing Li, Huan Ling, Seung Wook Kim, Karsten Kreis, Sanja Fidler, and Antonio Torralba. Bigdatasetgan: Synthesizing imagenet with pixel-wise annotations. In *CVPR*, 2022. 3, 29

[68] Zhiheng Li and Chenliang Xu. Discover the Unknown Biased Attribute of an Image Classifier. In *ICCV*, 2021. 8

[69] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *arXiv preprint arXiv:2305.13655*, 2023. 7

[70] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 15, 63

[71] Sebastian Nørgaard Llambias, Mads Nielsen, and Mostafa Mehdipour-Ghazi. Data augmentation-based unsupervised domain adaptation in medical imaging. In *arXiv preprint arXiv:2308.04395*, 2023. 25

[72] Justin Lo, Jillian Cardinell, Alejo Costanzo, and Dafna Sussman. Medical augmentation (med-aug) for optimal data augmentation in medical deep learning networks. In *Sensors*, 2021. 25

[73] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, 2018. 37, 75

[74] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 33

[75] Jinqi Luo, Zhaoning Wang, Chen Henry Wu, Dong Huang, and Fernando De La Torre. Zero-shot model diagnosis. In *CVPR*, 2023. 6, 8, 9, 62, 66

[76] Jinqi Luo, Zhaoning Wang, Chen Henry Wu, Dong Huang, and Fernando De La Torre. Zero-shot model diagnosis. In *CVPR*, 2023. 29

[77] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. Adversarial style mining for one-shot unsupervised domain adaptation. In *NeurIPS*, 2020. 78

[78] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *ICLR*, 2018. 3, 5, 8, 14

[79] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2022. 28

[80] Ramaravind K. Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *ACM FAccT*, 2020. 5, 8

[81] Nipun R. Navadia, Gurleen Kaur, Harshit Bhadwaj, Taranjeet Singh, Yashpal Singh, Indu Malik, Arpit Bhardwaj, and Aditi Sakalle. *Challenges and Opportunities for Deep Learning Applications in Industry 4.0.* Bentham Science Publishers, 2022. 1

[82] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. 28

[83] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 28

[84] Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. In *JAIR*, 2021. 27

[85] Curtis G. Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. In *NeurIPS*, 2021. 32

[86] OpenAI. Gpt-4 technical report. 2023. 1, 21, 60

[87] OpenAI. Introducing GPTs, Nov 2023. 1

[88] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In *ICCV*, 2021. 7, 10

[89] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021. 28

[90] Duo Peng, Qiuhong Ke, Yinjie Lei, and Jun Liu. Unsupervised domain adaptation via domain-adaptive diffusion. In *ITIP*, 2023. 29

[91] Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, and Kate Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *CVPRW*, 2018. 25

[92] Juan C Pérez, Motasem Alfarra, Ali Thabet, Pablo Arbeláez, and Bernard Ghanem. Towards characterizing the semantic robustness of face recognition. In *CVPR*, 2023. 8

[93] Viraj Prabhu, Sriram Yenamandra, Prithvijit Chattopadhyay, and Judy Hoffman. Lance: Stress-testing visual models by generating language-guided counterfactual images. *NeurIPS*, 2023. 8, 9

[94] Viraj Prabhu, Sriram Yenamandra, Prithvijit Chattopadhyay, and Judy Hoffman. Lance: Stress-testing visual models by generating language-guided counterfactual images. In *NeurIPS*, 2023. 29

[95] Haonan Qiu, Chaowei Xiao, Lei Yang, Xinchen Yan, Honglak Lee, and Bo Li. SemanticAdv: Generating Adversarial Examples via Attribute-conditioned Image Editing. In *ECCV*, 2020. 6, 8

[96] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 7, 27

[97] Aaron Raj. Ai in film industry: The world's first feature-length ai-generated film, Jan 2024. 1

[98] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 28

[99] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 38

[100] Pau Rodríguez, Massimo Caccia, Alexandre Lacoste, Lee Zamparo, Issam Laradji, Laurent Charlin, and David Vazquez. Beyond trivial counterfactual explanations with diverse valuable explanations. In *ICCV*, 2021. 8

[101] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2021. 7

[102] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 27, 28, 31

[103] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023. 28

[104] Shiori Sagawa, Pang Wei Koh, Tony Lee, Irena Gao, Sang Michael Xie, Kendrick Shen, Ananya Kumar, Weihua Hu, Michihiro Yasunaga, Henrik Marklund, Sara Beery, Etienne David, Ian Stavness, Wei Guo, Jure Leskovec, Kate Saenko, Tatsunori Hashimoto, Sergey Levine, Chelsea Finn, and Percy Liang. Extending the wilds benchmark for unsupervised adaptation. In *ICLR*, 2022. 25

[105] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022. 28

[106] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 2018. 37, 75

[107] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *SIGGRAPH*, 2022. 7

[108] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 25, 66

[109] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. InterFaceGAN: Interpreting the Disentangled Face Representation Learned by GANs. In *IEEE TPAMI*, 2020. 7

[110] Yujun Shen and Bolei Zhou. Closed-Form Factorization of Latent Semantics in GANs. In *CVPR*, 2021. 7

[111] Joonghyuk Shin, Minguk Kang, and Jaesik Park. Fill-up: Balancing long-tailed data with generative models. *arXiv preprint arXiv:2306.07200*, 2023. 29

[112] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *CVPRW*, 2023. 29

[113] Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. *ICLR*, 2020. 8, 11, 66

[114] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 7, 12, 28

[115] Tao Sun, Mattia Segu, Janis Postels, Yuxuan Wang, Luc Van Gool, Bernt Schiele, Federico Tombari, and Fisher Yu. Shift: A synthetic driving dataset for continuous multi-task domain adaptation. In *CVPR*, 2022. 25

[116] The White House. Executive order on the safe, secure, and trustworthy development and use of artificial intelligence, Oct 2023. 2

[117] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *ICLRW*, 2023. 29, 33, 34, 37, 40, 75

[118] Matthew Trager, Pramuditha Perera, Luca Zancato, Alessandro Achille, Parminder Bhatia, and Stefano Soatto. Linear spaces of meanings: Compositional structures in vision-language models. In *ICCV*, 2023. 30

[119] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. In *JMLR*, 2008. 39

[120] Joshua Vendrow, Saachi Jain, Logan Engstrom, and Aleksander Madry. Dataset interfaces: Diagnosing model failures using controllable counterfactual generation. *arXiv preprint arXiv:2302.07865*, 2023. 8, 29

[121] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. *p+*: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 28

[122] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 2017. 8

[123] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE TPAMI*, 2019. 64

[124] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004. 11

[125] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *ICCV*, 2021. 18, 64

[126] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J. Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *ICCV*, 2021. 25

[127] Chen Henry Wu and Fernando De la Torre. A latent space of stochastic diffusion models for zero-shot image editing and guidance. In *ICCV*, 2023. 28

[128] Zongze Wu, Dani Lischinski, and Eli Shechtman. StyleSpace Analysis: Disentangled Controls for StyleGAN Image Generation. In *CVPR*, 2021. 7, 10

[129] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE TPAMI*, 2022. 10

[130] Chaowei Xiao, Bo Li, Jun-yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating Adversarial Examples with Adversarial Networks. In *IJCAI*, 2018. 8

[131] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, 2015. 32

[132] Sherry Yang, Ofir Nachum, Yilun Du, Jason Wei, Pieter Abbeel, and Dale Schuurmans. Foundation models for decision making: Problems, methods, and opportunities. *arXiv preprint arXiv:2303.04129*, 2023. 1

[133] Jianhao Yuan, Francesco Pinto, Adam Davies, Aarushi Gupta, and Philip Torr. Not just pretty pictures: Text-to-image generators enable interpretable interventions for robust representations. In *arXiv preprint arXiv:2212.11237*, 2022. 29

[134] Shih-Cheng Huang Kuan-Chieh Wang James Zou Serena Yeung Yuhui Zhang, Jeff Z. HaoChen. Diagnosing and rectifying vision models using language. In *ICLR*, 2023. 8

[135] Mehdi Zemni, Mickaël Chen, Éloi Zablocki, Hédi Ben-Younes, Patrick Pérez, and Matthieu Cord. Octet: Object-aware counterfactual explanations. In *CVPR*, 2023. 8, 11

[136] Cheng Zhang, Xuanbai Chen, Siqi Chai, Chen Henry Wu, Dmitry Lagun, Thabo Beeler, and Fernando De la Torre. Iti-gen: Inclusive text-to-image generation. In *ICCV*, 2023. 28

[137] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 25

[138] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 7

[139] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 28

[140] Tianjun Zhang, Yi Zhang, Vibhav Vineet, Neel Joshi, and Xin Wang. Controllable text-to-image generation with gpt-4. *arXiv preprint arXiv:2305.18583*, 2023. 7

[141] Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. Datasetgan: Efficient labeled data factory with minimal human effort. In *CVPR*, 2021. 26, 29

[142] Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data. *ICMLW*, 2023. 29

[143] Jingyuan Zhu, Huimin Ma, Jiansheng Chen, and Jian Yuan. Domainstudio: Fine-tuning diffusion models for domain-driven image generation using limited data. *arXiv preprint arXiv:2306.14153*, 2023. 29

[144] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 28

[145] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks. In *ICLR*, 2022. 28

Appendix A

Model Diagnosis

A.1 Multi-Direction Edit Vector Optimization

Algorithm 1 shows the pseudo-code of the multi-direction edit vector optimization process. We start with initializing all k edit vectors as a list of vectors Δs. Then for each iteration, we first generate a random image x and then evaluate the effectiveness of each edit vector $\Delta s[i]$ on this image. The most effective vector is then optimized on the current image x while the other edit vectors remain unchanged. Under this design, for similar attribute changes, the same most effective edit vector tends to be optimized further and further. For distinct semantic perturbations, other edit vectors may be selected and optimized to represent the new change. Hence after convergence, the multiple edit vectors are likely to capture more comprehensive edits.

To visualize what each edit vector captures, we optimize four latent vectors when diagnosing a perceived gender classifier as an example. Fig. A.1 visualizes the semantic changes captured in each edit. Among the edit direction interpolations on the four random images, we can observe that edit 1 increases image contrast and changes skin tone, edit 2 increases image exposure, edit 3 adds smiles, while edit 4 removes smiles. This figure shows that Algorithm 1 effectively optimizes multiple edit vectors into different semantic perturbations. Moreover, different faces require different semantic edits to mislead model predictions, hence increasing failure coverage.

Algorithm 1 Multi-Direction Edit Optimization

Input: k - Number of edit vectors
n - Number of training samples
$G(z)$ - Generative backbone

for i in $[0, k]$ **do**
  Initialize each edit vector $\Delta s[i] \sim \mathcal{N}(0, 0.01)$
for j in $[0, n]$ **do**
  $z \leftarrow$ samples from generative latent space
  $x \leftarrow G(z)$ ▷ *Generate the original image*
  for each $\Delta s[i]$ **do**
    $\hat{x}_i \leftarrow G(z + \Delta s[i])$ ▷ *Generate edited image*
    $d[i] \leftarrow \mathcal{L}_{\text{target}}(\hat{x}_i)$ ▷ *Compute effectiveness*
  $i \leftarrow \text{argmax}_i \, d[i]$ ▷ *Select the most effective edit*
  $\Delta s[i] \leftarrow \Delta s[i] - \eta \nabla_{\Delta s[i]} \mathcal{L}$ ▷ *Optimize with SGD*

Output: Δs

Immerse yourself into the role of a Model Diagnosis Expert (MDE) AI. MDE, as the name suggests, is an expert in Trustworthy Machine Learning that has knowledge in analyzing deep models' robustness, fairness, and interpretability. You can tell me the common causes of deep learning model failures. You can think carefully and exhaustively about what the relevant attributes of a given task are. Next, I'll give you instructions. We are analyzing what kinds of attributes in CelebA are most influential/sensitive to the classifier's decision. You must try your best and think carefully to propose an exhaustive list of attributes that are relevant to our task in this domain. The results need to be a list of lists where each list corresponds to the different options/values for an attribute. The attributes should be as specific as possible and the values should be nouns such that they fit in the sentence "a face with ___". For example, instead of a "hair" attribute, you should list "hairstyles", "hair colors", etc. Make the list as long as needed to be extremely comprehensive. Do not use "others" or "etc." but list all options out as much as you can. Also, avoid yes and no questions. Remember we want adversarial attributes that the classifier may be sensitive to.

Table A.1: **Prompt for GPT-4 to populate candidates for human face domain.**

A.2 Text Attribute Candidates

In the counterfactual analysis module, among many sources proposed in Sec. 2.3.2, for efficiency and simplicity, we leveraged GPT-4 [86] to propose our attribute candidate list. An example prompt we used for human face attribute candidates is in Tab. A.1. The full candidate list returned from the above prompt is in **??**. This list is a relatively comprehensive collection of relevant attributes to our task.

Figure A.1: **Visualization of multiple edit vectors optimized from Algorithm 1.** For each example, the middle images are the unedited original generation; the right and left images are the images edited by gradually adding or subtracting the edit vector. This figure demonstrates that multiple edit vectors are successfully optimized to capture different semantic changes. Moreover, different original images can be susceptible to different semantic edits (blue and orange boxes indicate counterfactual pairs), hence increasing our diagnosis coverage.

A.3 Iterative Attribute Selection

In Sec. 2.3.2, the iterative attribute selection algorithm for our counterfactual analysis module is defined as Algorithm 2. Recall that the score metrics S_{sim} and S_{uni} are defined in Sec. 2.3.2.

Algorithm 2 Iterative Attribute Selection

Input: k - Number of top attributes to discover
S_a - Set of all attribute candidates

$S_r \leftarrow \emptyset$ ▷ *The set of select attributes*
Sim $\leftarrow \emptyset$ ▷ *Similarity score dictionary*
Uni $\leftarrow \emptyset$ ▷ *Uniqueness score dictionary w.r.t.* S_r

function SELECTNEXTATTR(S_r)
    for a_i in S_a **do**
        Uni$[a_i] \leftarrow S_{\text{uni}}(a_i, S_r)$
    return argmax_{a_i} Sim$[a_i]$ · Uni$[a_i]$

for a_i in S_a **do**
    Sim$[a_i] \leftarrow S_{\text{sim}}(a_i)$
while $|S_r| < k$ **do**
    $a_{\text{next}} \leftarrow$ SELECTNEXTATTR(S_r)
    $S_r \leftarrow S_r \cup \{a_{\text{next}}\}$

Output: S_r

A.4 Counterfactual Effectiveness

Sec. 2.3.3 describes one direct application of our diagnosis is to integrate UMO into a counterfactual training process, where tailored counterfactual training data are produced (as shown in Fig. A.2) and added into each adversarial training step. In this section, we quantify the effectiveness of UMO-integrated counterfactual training.

To evaluate the counterfactual robustness of an improved model, we resort to the Flip Resistence (FR) metric defined in ZOOM [75] which is the percentage of images where counterfactual attacks are ineffective. Given a base model, we improve it by ZOOM and UMO respectively. Then we evaluate against three metrics: CelebA classification accuracy, FR-25 and FR-100, where 25 and 100 denote the iteration steps during counterfactual optimization, *i.e.* different extent of counterfactual optimization.

The experiment is repeated on two models: a perceived age classifier and a big-lips classifier. The results are shown in Tab. A.3. In both tasks, all of the base model, ZOOM-improved model and UMO-improved model achieves at-par performance on the original CelebA test set. However, when diagnosed by UMO, our counterfactual optimization can easily find semantic edits that flip the model prediction for the base

Figure A.2: **Counterfactual training samples.** These images are generated during the counterfactual training and merged into the training data to robustify the base model, which results in our improved model in Tab. A.3.

model and model improved by ZOOM. Moreover, the model improved by UMO stays robust against open-domain counterfactual attacks.

A.5 Dataset Diagnosis

In addition to our diagnosis verification experiment in Sec. 2.4.1, we also applied UMO to classifiers trained on the full CelebA [70] dataset. Since these classifiers can be seen as a compressed representation of the dataset, by diagnosing the model, we may uncover issues in the dataset as well. As we showed the correlation between "Brown Hair" and "Young" in Sec. 2.4.1, this section focuses on exploring more biases in the CelebA dataset via our diagnosis of the model.

We studied biases against two attributes in CelebA perceived age (Young) and perceived gender (Male). Hence we trained one classifier on each attribute of interest. By diagnosing these classifiers, we obtained the top-10-matched CelebA attributes with their similarity scores presented as the line plot in Fig. A.3. Then we explore each attribute in the dataset by counting the co-occurrences of the attribute and the main class, shown as the bar plots in Fig. A.3. We observe that all diagnosed attributes have imbalanced distributions in the dataset. For instance, regarding the Senior/Young class, the majority of samples with "Rosy Cheeks", "Brown Hair", "Wearing Necklace", etc. are also labeled as "Young", which introduces spurious

correlations to the classifier. On the other hand, considering the perceived gender classifier, although samples with "Straight Hair", "Narrow Eyes", "Black Hair" have even distributions between the labels of "Male" and "Female", the majority of samples labeled as "Male" do not contain these attributes. Hence UMO is capable of indirectly uncovering dataset biases via the lens of model diagnosis.

A.6 More Counterfactual Visualizations

In this section, we show more counterfactual explanations in each of our experiments in Sec. 2.4. Figs. A.4 to A.6 show more counterfactual pairs illustrating edits that lead to model failures in CelebA. These diagnosed models are resnet50 [37] networks trained on imbalanced datasets as described in Sec. 2.4.1. Moreover, quantitative evaluations are conducted on our counterfactual images in the CelebA setting in Tab. A.4. Since our method directly integrates off-the-shelf pretrained generative models without any fine-tuning, it is not our effort to improve the generation quality. Nonetheless, Tab. A.4 shows that our edits do not distort the synthetic image quality, which remained within a reasonable range. Fig. A.7 visualizes counterfactual edits for our cat/dog classifier in Sec. 2.4.2 in addition to Fig. 2.5. This resnet50 classifier is trained on the AFHQ dataset [17]. Figs. A.8 and A.9 provide more visual counterfactual explanations under the task of segmentation and keypoint detection, extending Fig. 2.7. As described in Sec. 2.4.3, the segmentation model is a resnet50 pretrained on ImageNet [22] and the HRNetV2 [123] keypoint detector is trained on the FITYMI dataset [125]. These additional figures exhibit more visual support to our diagnosis.

Attributes	Values
Hairstyles	short, medium, long, curly, straight, wavy, braided, bald, mohawk, bun, pixie cut, dreadlocks, undercut, pompadour, buzz cut, side part, bob cut, cornrows
Hair Colors	black, brown, blonde, red, gray, white, pink, blue, purple, green, multi-color
Facial Hair Styles	beard, goatee, mustache, sideburns, clean-shaven, stubble, handlebar, soul patch, five o'clock shadow, full beard
Eye Shapes	almond, round, monolid, hooded, upturned, downturned
Nose Shapes	Roman, snub, Greek, aquiline, hawk, button
Lip Types	full, thin, heart-shaped, wide
Face Shapes	oval, round, square, heart-shaped, diamond-shaped, rectangular
Eyebrows	thick, thin, unibrow, arched, straight
Eye Colors	blue, brown, green, gray, hazel, black, amber
Glasses Types	reading, sunglasses, aviator, cat-eye, round, square, rimless
Makeup	eyeliner, eyeshadow, lipstick, mascara, blush, foundation, contouring
Accessories	earrings, necklace, hat, cap, headscarf, headband, bandana, tie, bow tie, septum piercing, lip piercing, eyebrow piercing
Skin Types	light, medium, dark, freckled, tanned, pale
Skin Conditions	acne, scars, birthmarks, vitiligo, rosacea, wrinkles
Facial Expressions	smiling, frowning, surprised, neutral, angry, crying, winking
Age Categories	child, teenager, adult, elderly
Hair Texture	frizzy, oily, dry, shiny, coarse, smooth
Ear Types	big, small, pointed, flat, protruding, pierced
Cheek Characteristics	high cheekbones, low cheekbones, chubby, hollow
Chin/Jaw Attributes	double chin, prominent jawline, weak jawline, square jaw, round jaw
Forehead	high, low, wide, narrow
Teeth	straight, crooked, missing, gap, braces, white, yellow
Cosmetic Alterations	nose job, lip fillers, botox, cheek fillers, chin augmentation
Eyelashes	long, short, false
Eyewear	contact lenses, monocle, pince-nez
Headwear	beanie, beret, baseball cap, hijab, turban, fedora, helmet, headwrap
Brow Treatments	microblading, eyebrow tinting, eyebrow piercing
Facial Piercings	cheek, chin, dermal
Hair Treatments	perms, straightening, extensions, highlights, lowlights
Facial Symmetry	symmetrical, asymmetrical
Photo Lighting	soft, harsh, backlit, frontlit, side-lit
Photo Angles	front-facing, profile, three-quarters
Background	indoor, outdoor, neutral, busy

Table A.2: **Examples of attribute candidates proposed by GPT-4.** Using the prompt in Tab. A.1, GPT-4 returned this list of attribute candidates relevant for models trained on CelebA, which is relatively comprehensive and sufficient for our task.

Classifier	Metric (↑, %)	Base	ZOOM [75]	Ours
Perceived Age	CelebA Accuracy	86.70	87.31	86.23
	FR-25	0.78	8.59	**98.44**
	FR-100	0.00	0.00	**96.09**
Big Lips	CelebA Accuracy	70.00	69.72	69.97
	FR-25	0.00	17.97	**100.00**
	FR-100	0.00	0.00	**97.66**

Table A.3: **Counterfactual training evaluation**. This table assesses the robustness of the counterfactual training with UMO against two baselines in two classification tasks. We show significant robustness of the model after our improvement while maintaining at-par performance on the regular test set.

	FID [41]↓	Face Sim [113]↑	Aesthetics [108]↑
ZOOM [75]	41.2	100%	4.68
Ours	41.3	99.6%	4.91

Table A.4: **Quantitative evaluation of our counterfactual generation on CelebA task**. We compare UMO against ZOOM [75]. The metrics indicate that our learned edit vectors preserve a good quality of the generated counterfactual images.

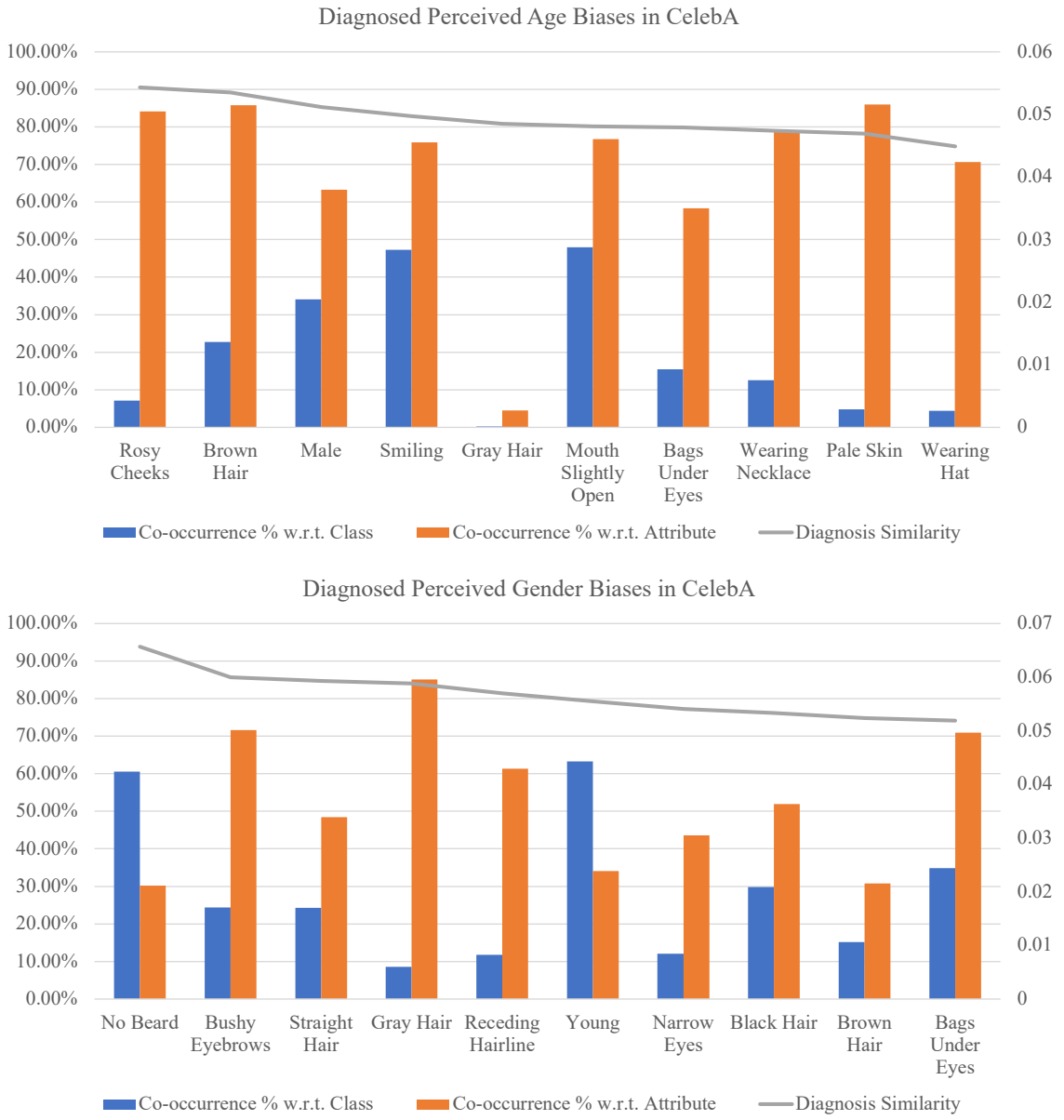

Figure A.3: **Co-occurrence statistics of diagnosed CelebA attributes.** We indirectly diagnosed biases and correlations in the CelebA dataset by diagnosing a model trained on CelebA. Using CelebA attributes as text candidates, we diagnosed a perceived age classifier and a perceived gender classifier and reported the top-10-matched attributes. The matching similarity score is shown as the gray line. The blue bar indicates the percentage of co-occurrences in the main class; the orange bar indicates the percentage of co-occurrences in the potentially correlated attributes. This figure shows that each discovered attribute indeed has an imbalanced distribution in the dataset.

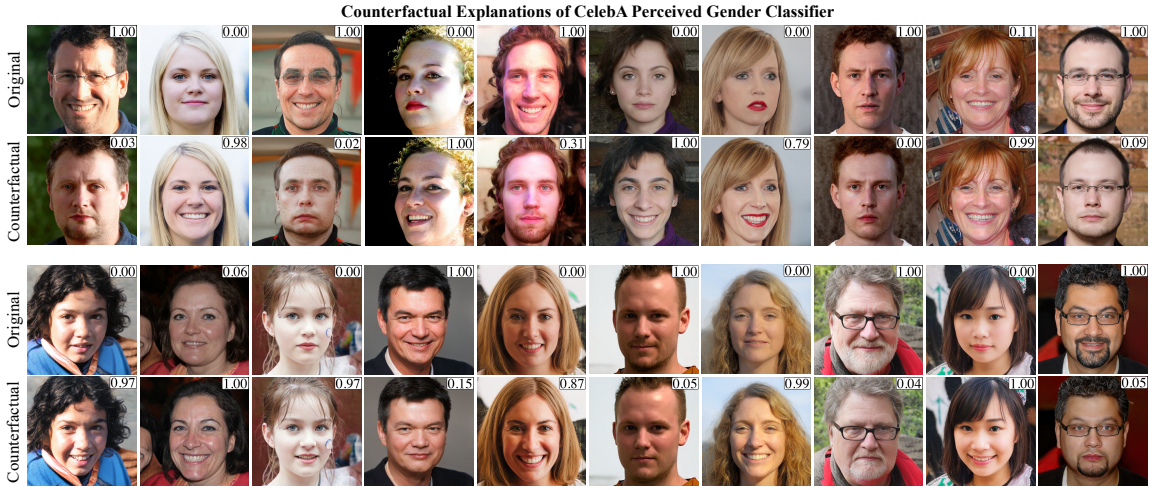

Figure A.4: **More counterfactual pairs for the perceived gender classifier.** We provide more visualizations for the perceived gender classifier in Fig. 2.3. Given random latent vectors, we generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The number on the top-right indicates the prediction score (0-Perceived Female / 1-Perceived Male). Our counterfactual explanation visualizes attributes that mislead model predictions.

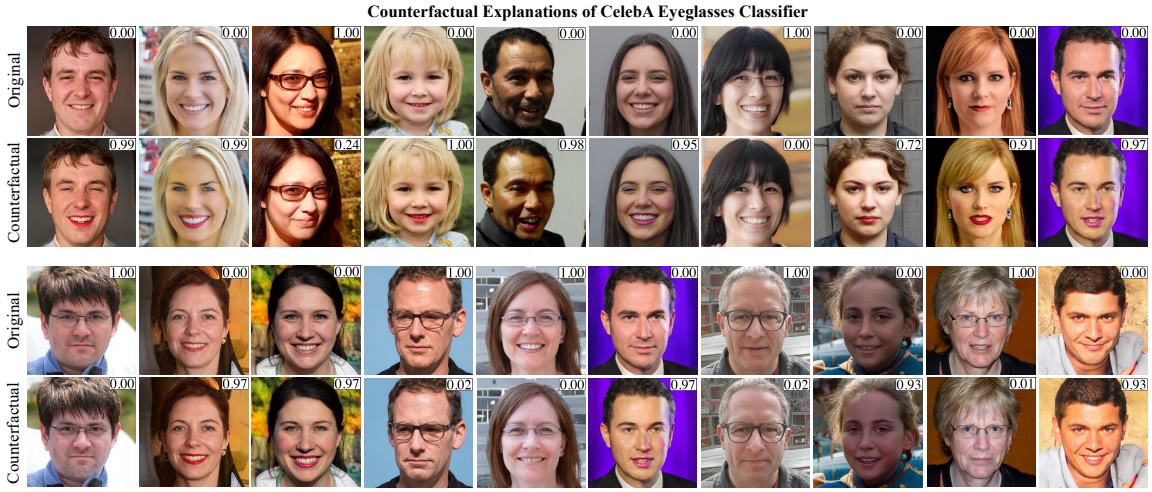

Figure A.5: **More counterfactual pairs for the eyeglasses classifier.** We provide more visualizations for the eyeglasses classifier in Fig. 2.3. Given random latent vectors, we generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The number on the top-right indicates the prediction score (0-No Eyeglasses / 1-Wearing Eyeglasses). Our counterfactual explanation visualizes attributes that mislead model predictions.

Figure A.6: **More counterfactual pairs for the perceived age classifier.** We provide more visualizations for the perceived age classifier in Fig. 2.3. Given random latent vectors, we generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The number on the top-right indicates the prediction score (0-Perceived Senior / 1-Perceived Young). Our counterfactual explanation visualizes attributes that mislead model predictions.

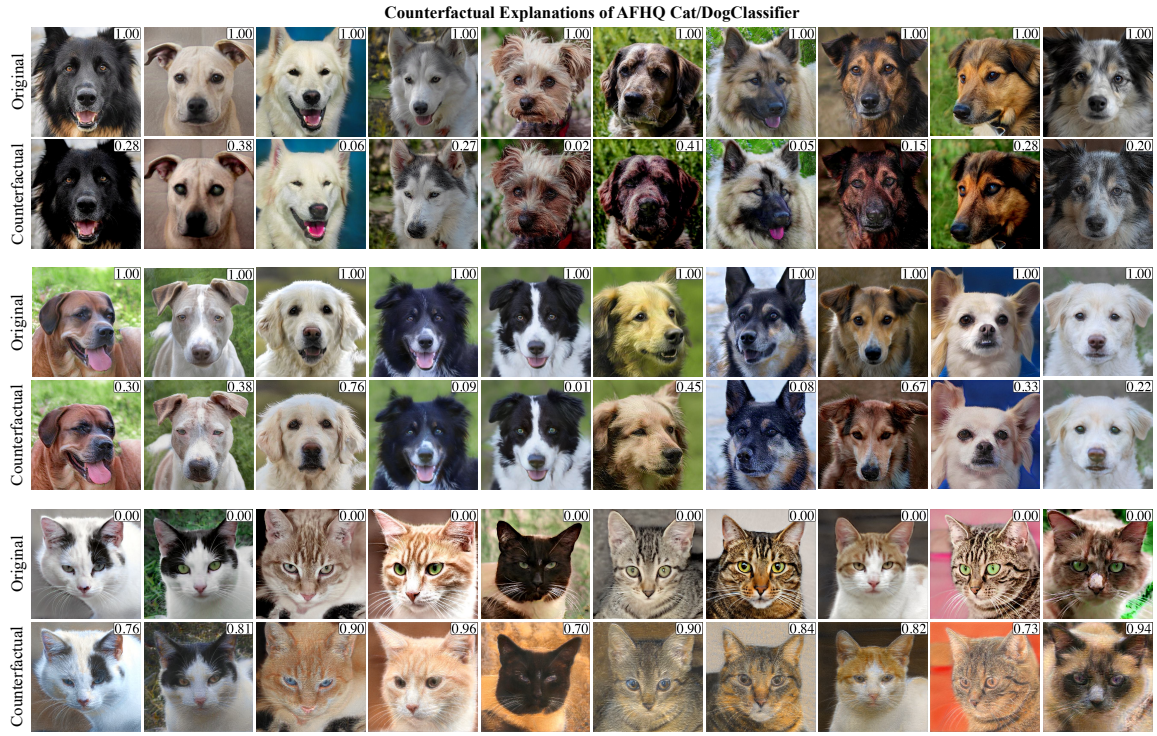

Figure A.7: **More counterfactual pairs for the cat/dog classifier.** We provide more visualizations for the cat/dog classifier in Fig. 2.5. We generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The number on the top-right indicates the prediction score (0-Cat / 1-Dog). Our counterfactual explanation visualizes attributes that mislead model predictions.

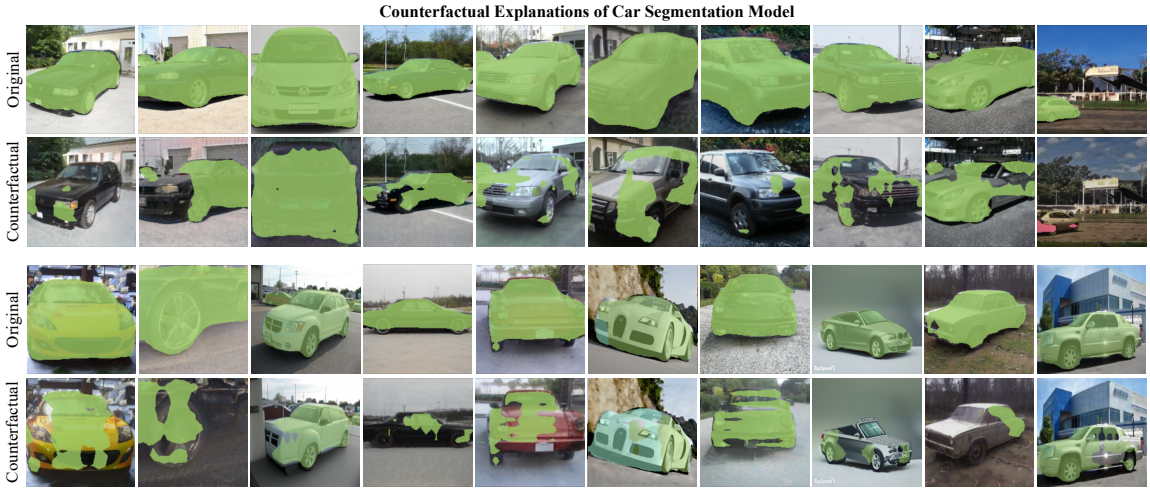

Figure A.8: **More counterfactual pairs for the car segmentation model.** We provide more visualizations for the car segmentation in Fig. 2.7. We generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The green mask indicates predicted areas for cars. Our counterfactual explanation visualizes attributes that mislead model predictions.

Figure A.9: **More counterfactual pairs for the keypoint detection model.** We provide more visualizations for the keypoint detector in Fig. 2.7. We generate the original and perturbed (counterfactual) image pairs where small semantic changes flip the predicted class. The red dots indicate predicted facial keypoints. Our counterfactual explanation visualizes attributes that mislead model predictions.

Appendix B

Model Improvement

B.1 Domain Gap Extraction

In this section, we explore more details in the domain gap extraction process, including different algorithms to capture the domain gap representation and the impact of sample sizes available for the extraction step.

B.1.1 Extraction Methods

In Sec. 3.3.1, we described two options for our domain gap extraction algorithms: the difference of means and the PCA-based method. For analysis, we qualitatively compare DomainNet (Real → Painting) generations using the two methods. We visualize the impact of different methods in Fig. B.2. The results show that the difference of means yields better adaptation effectiveness, *i.e.* more aligned to target domains, than the PCA-based method. We adopt the difference of means as our domain gap representation for the following experiments.

B.1.2 Impact of Target Set Size

Besides the extraction algorithm, in our few-shot setting, the impact of different numbers of target samples available is also important to study. We evaluated the performances of our synthetic data generated with domain gap embeddings from different numbers of target samples. For analysis, we considered the first 20 classes in DomainNet (Real → Painting) and evaluated the performance on the 20-class classification task. We randomly sampled the same number of images per class from

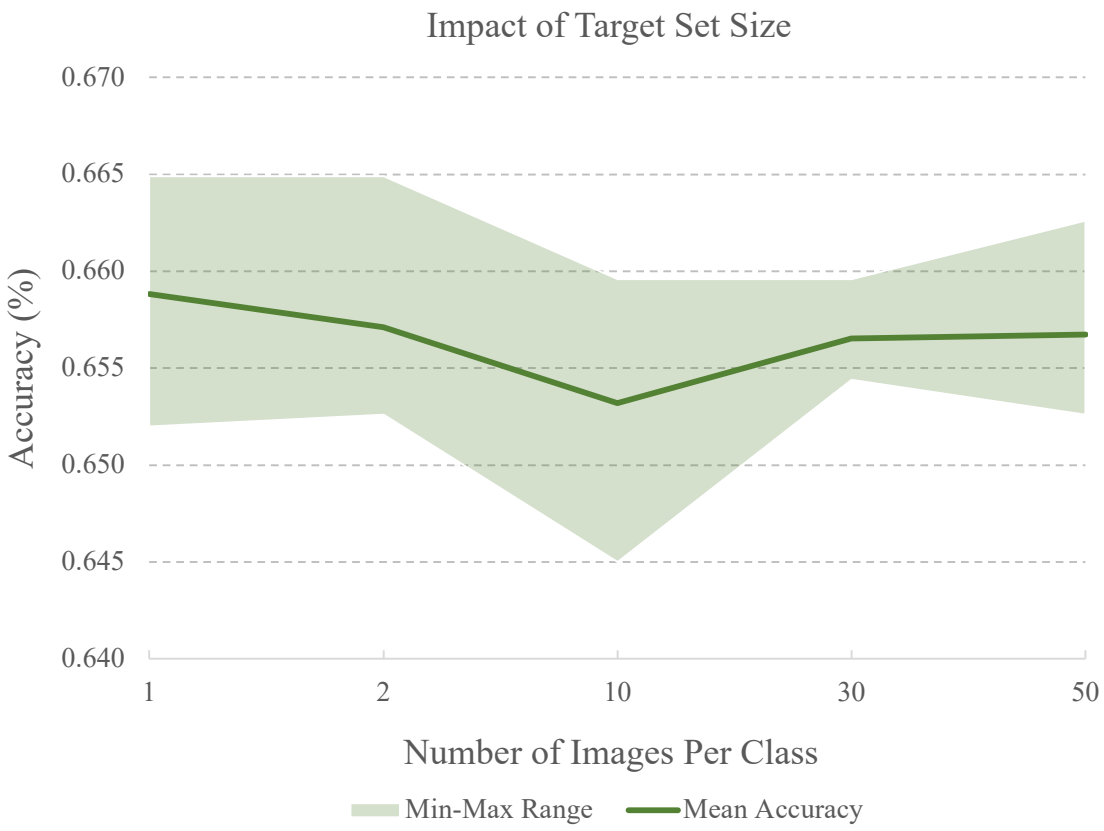

Figure B.1: **Line plot of the impact of different target data sizes.** We evaluated the performance on our 20-class DomainNet (Real → Painting) classification task. The results illustrate that using as little as one image per class from the target distribution is sufficient for DoGE to generate effective training data.

the source and target distributions, from 1, 2, 10, 30 to 50. Fig. B.1 shows using as little as one image per class (20 images) from the target distribution is as effective as using 50 images per class (1000 images).

B.2 Data Cleaning Algorithms

To further explain the training-time confidence-based data cleaning process in Sec. 3.3.3, we include Algorithm 3. Given a model $f(x)$ trained on original training data, we adapt it by fine-tuning on our synthetic dataset. During fine-tuning, for each batch of data in the original training set, we pre-computed the image augmentations with

Figure B.2: **Synthetic data from different domain gap extraction algorithms.** (a) Real source images from DomainNet were converted into Painting using (b) PCA and (c) the difference of means, shown accordingly. These examples illustrate that (c) is more effective than (b) in augmenting source (Real domain) images into the target distribution (Painting domain).

Algorithm 3 Confidence-Based Data Cleaning

Input: $G(x)$ - Our DoGE data augmenter
$f(x)$ - The downstream task model to improve
(X, Y) - The source training set data and labels
t - Threshold for confidence-based filtering

for batch b with label y_b in (X, Y) **do**
  $\hat{b} \leftarrow G(b)$ ▷ *Augment source data to target domain*
  for synthetic sample $\hat{x}$ with label y in $\hat{b}$ **do**
    $\hat{p} \leftarrow \text{argmax}\, f(\hat{x})$ ▷ *Model prediction*
    $c \leftarrow \max f(\hat{x})$ ▷ *Model confidence*
    if $\hat{p} \neq y$ and $c \geq t$ **then**
      discard $\hat{x}$ from $\hat{b}$
  $f \leftarrow \text{AdamW}(\mathcal{L}(f(\hat{b}))$ ▷ *Update model*

Output: $f(x)$ - The adapted and improved model

DoGE and denote the corresponding augmented batch as $\hat{b}$. For each generation $\hat{x} \in \hat{b}$ with the original label y, we use the current model to predict the label $\hat{p} = f(\hat{x})$ and compute the confidence as the maximum softmax score among all classes $c = \max f(\hat{x})$. Then, we ignore $\hat{x}$ from this training batch if the prediction is wrong *i.e.* $\hat{p} \neq y$ with high confidence c over a certain threshold t. After we filter the entire batch as above, then we fine-tune the model on the cleaned batch.

	UDA methods	Test Acc (%) —	+ DA-Fusion [117]	+ DATUM [5]	+ DoGE
Real → **P**ainting	BSP [16]	46.76	46.78	41.89	**47.34**
	DANN [29]	47.01	48.83	42.73	**49.68**
	CDAN [73]	51.66	51.91	49.87	**52.11**
	MCD [106]	50.88	50.99	49.71	**52.14**
	MCC [53]	50.08	50.42	49.31	**52.95**
	MemSAC [55]	52.27	53.26	50.32	**54.16**
Real → **C**lipart	BSP [16]	46.78	45.11	39.43	**46.79**
	DANN [29]	**49.80**	47.82	42.70	48.11
	CDAN [73]	53.93	54.11	50.54	**54.53**
	MCD [106]	51.42	50.79	50.01	**54.02**
	MCC [53]	50.61	49.27	48.10	**51.99**
	MemSAC [55]	54.34	54.59	51.10	**55.35**
Real → **S**ketch	BSP [16]	36.47	36.81	28.38	**38.49**
	DANN [29]	38.72	38.45	36.13	**40.21**
	CDAN [73]	42.60	42.23	39.65	**43.00**
	MCD [106]	39.25	38.07	39.19	**42.78**
	MCC [53]	34.38	33.31	33.06	**37.23**
	MemSAC [55]	41.74	40.42	36.54	**43.23**

Table B.1: **Test Accuracy of UDA methods on the DomainNet problem.** We evaluated existing UDA methods with and without synthetically supplemented training data; +DA-Fusion, +DATUM and +DoGE denote the methods used for the generation. This table shows that DoGE, while being compatible with and complementary to UDA methods, is also more effective than the competing methods.

B.3 Real-Synthetic Mixing Ratio

While the above data cleaning process filters out poor-quality samples and improves the usefulness of synthetic data, the effectiveness of our data is also dependent on how we leverage them to fine-tune downstream task models. One important decision is, when fine-tuning task models, how to take the most advantage of synthetic generations and the high-quality original training data. Hence we study the impact of various data mixing ratios during task model fine-tuning. For analysis, we used the first 20 classes in DomainNet (Real → Painting) and evaluated the performance on the 20-class classification task. We changed the ratio of synthetic to real images in the training dataset from 1:1, 1:5 to 1:10. Fig. B.3 shows that expanding the dataset by as little as 10% can be as effective as adding 100% more data.

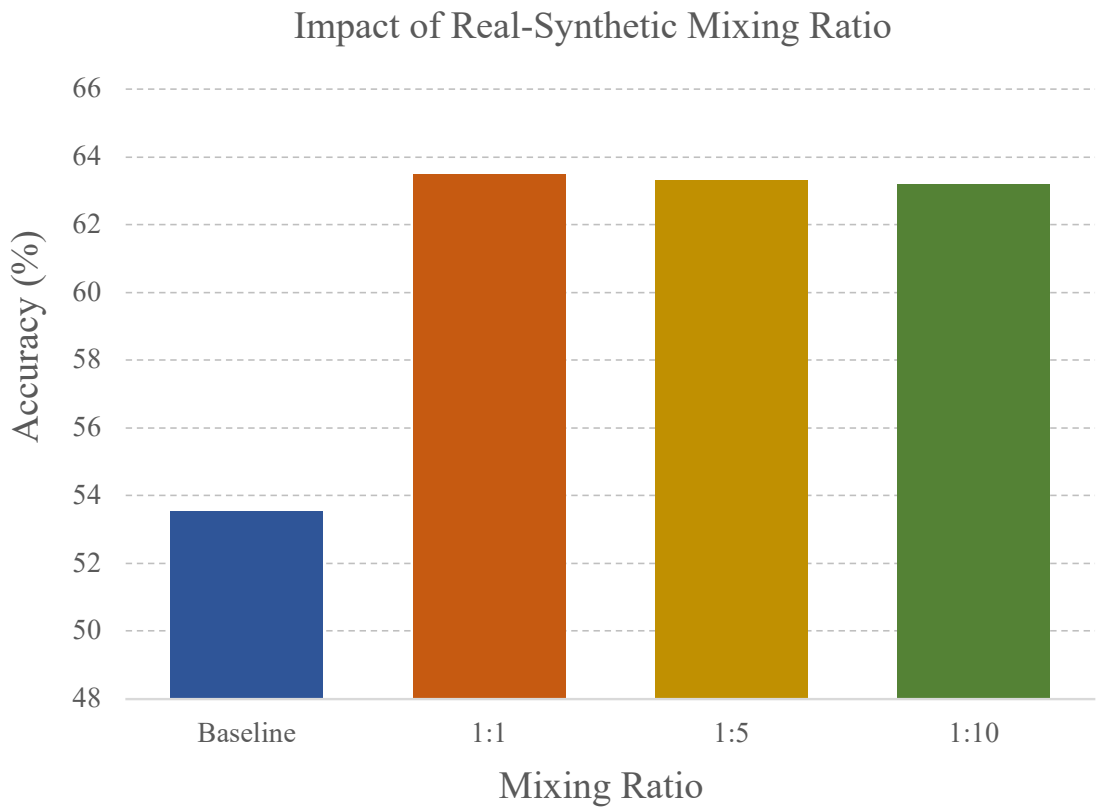

Figure B.3: **Bar plot of the impact of different data mixing ratios.** We evaluated the performance on our 20-class DomainNet (Real → Painting) classification task. The results show that augmenting as little as 10% of the training data is sufficient to improve downstream task model performances.

B.4 Complete UDA-Based Comparison

This section extends the brief experiment (Tab. 3.5) in Sec. 3.4.3, which shows that DoGE is compatible and complementary to existing UDA methods. We show the full UDA-based evaluations in Tab. B.1, where we also compare against other baselines DA-Fusion and DATUM on two more tasks. DoGE successfully improved and surpassed other baseline UDA evaluations in 17 out of 18 experiments.

B.5 More Method Ablation

To further isolate the effectiveness of and improvement from our domain gap embeddings, this section shows more ablation studies around CLIP and the generation pipeline.

	Acc↑		Acc↑	FID↓
zero-shot CLIP	53.53	Ours (noise)	38.97	39.36
finetuned CLIP	72.77	Ours (DoGE)	**44.00**	**18.25**

Table B.2: Evaluation on DomainNet (Real→Painting). (Left) We evaluated the zero-shot CLIP classifier and finetuned it on our synthetic dataset to show the effectiveness of our synthetic data. (Right) We compared embedding augmentation between noises and DoGE to demonstrate the performance gain.

B.5.1 Improvement on Top of CLIP

One foundation of the success of DoGE is the vast generalization capability and knowledge base in the CLIP latent space. However, in this section, we show that off-the-shelf CLIP is not sufficient against domain shifts. We focused on the DomainNet Real→Painting experiment setup in Sec. 3.4.3 and evaluated the zero-shot CLIP classifier against the CLIP classifier finetuned with our synthetic dataset in Tab. B.2 left. We can see that our synthetic dataset can effectively improve the zero-shot CLIP classifier further in domain shifts.

B.5.2 Domain Gap Embeddings Isolation

This section demonstrates the effectiveness of DoGE by isolating the Domain Gap Embeddings from the rest of the generation pipeline. Specifically, in the same DomainNet Real→Painting experiment setup in Sec. 3.4.3, we generated and evaluated two sets of synthetic datasets. One dataset generation used the default DoGE pipeline and the other replaced the injected domain gap embeddings with small random noises while keeping the rest of generation pipeline the same. Then we evaluated these datasets in terms of FID and classification accuracies by finetuning. As shown in Tab. B.2 right, using our domain gap embedding improves both FID and finetuning performance, demonstrating the effectiveness of DoGE.

B.6 Comparison to Style Transfer

Given the settings and method of DoGE, it may appear as a style transfer method. However, our goal, which is generative semantic data augmentation, is more than just style transfer. Similar to previous literature cited in the related works in Sec. 3.2, our method is designed for any kind of semantic data augmentation within CLIP's representation capacity rather than style transfer only. The first experiment in Sec. 3.4.2 shows our effectiveness in improving the subpopulation shift problem with object changes. Such augmentations (e.g., adding/removing eyeglasses in Fig. 3) are not regular style transfer tasks. Moreover, we can solve style transfer problems in a training-free and diverse fashion.

Nonetheless, existing style transfer methods [77, 30] are effective in many of our experiments and are important baselines to evaluate against. In Tab. B.3, we evaluate against other style transfer methods on our GTA→CityScapes experiment in Sec. 3.4.3. The table shows that our generalized method is as performant as style transfer methods.

	[77]	[30]	Ours
mIoU	44.5	55.37	**57.30**

Table B.3: Evaluation on GTA→CityScapes adaptation task.

B.7 More Visualizations

In this section, we present more samples of our generation that were briefly shown in Sec. 3.4. The generation setup is the same as mentioned in Sec. 3.4.1. For each task, we choose the difference of means as our domain gap representation. Except for synthetic CelebA data generation, we enable our ControlNet integration in every other task.

B.7.1 Imbalanced CelebA Classification

Fig. B.4 presents more generated samples for our CelebA experiment in Sec. 3.4.2. Recall that in this subpopulation shift scenario, the source and target distribution differ by the semantic change of adding/removing eyeglasses in perceived female/male

classes, as shown in the top-left corner of Fig. B.4. The rest of the figure shows more synthetic data in both classes augmented by our pipeline, *i.e.* males without eyeglasses and females with eyeglasses.

B.7.2 DomainNet Domain Adaptation

This section presents more visual examples of our DomainNet synthetic data used in Sec. 3.4.3. Figs. B.5 to B.7 display more of our generation from Real domain to Painting, Sketch, and Infograph domains. These data were used to improve classification model performance in our evaluations. Along with the reference target image on the left-most column, these figures demonstrate the quality and usefulness of our synthetic data.

B.7.3 FMoW Domain Adaptation

To extend the examples of synthetic FMoW data in Fig. 3.5, more samples are shown in Fig. B.8. As described in Sec. 3.4.3, we generate recent satellite images from an older period. Fig. B.8 lists 10 categories of land use and our generated data in each category. The figure illustrates our capability to generate high-quality satellite images.

B.7.4 GTA $\rightarrow$ CityScapes Segmentation

This section shows more generated data used in Sec. 3.4.3. The original training set is the GTA5 dataset and the target domain contains realistic driving scenes in CityScapes. Fig. B.9 shows more synthetic examples from DoGE. Since the segmentation map is available, we also show the control maps leveraged during the generation with ControlNet enabled. As the figure shows, the generated image is able to maintain the same image structure honoring the edge and segmentation mask constraints.

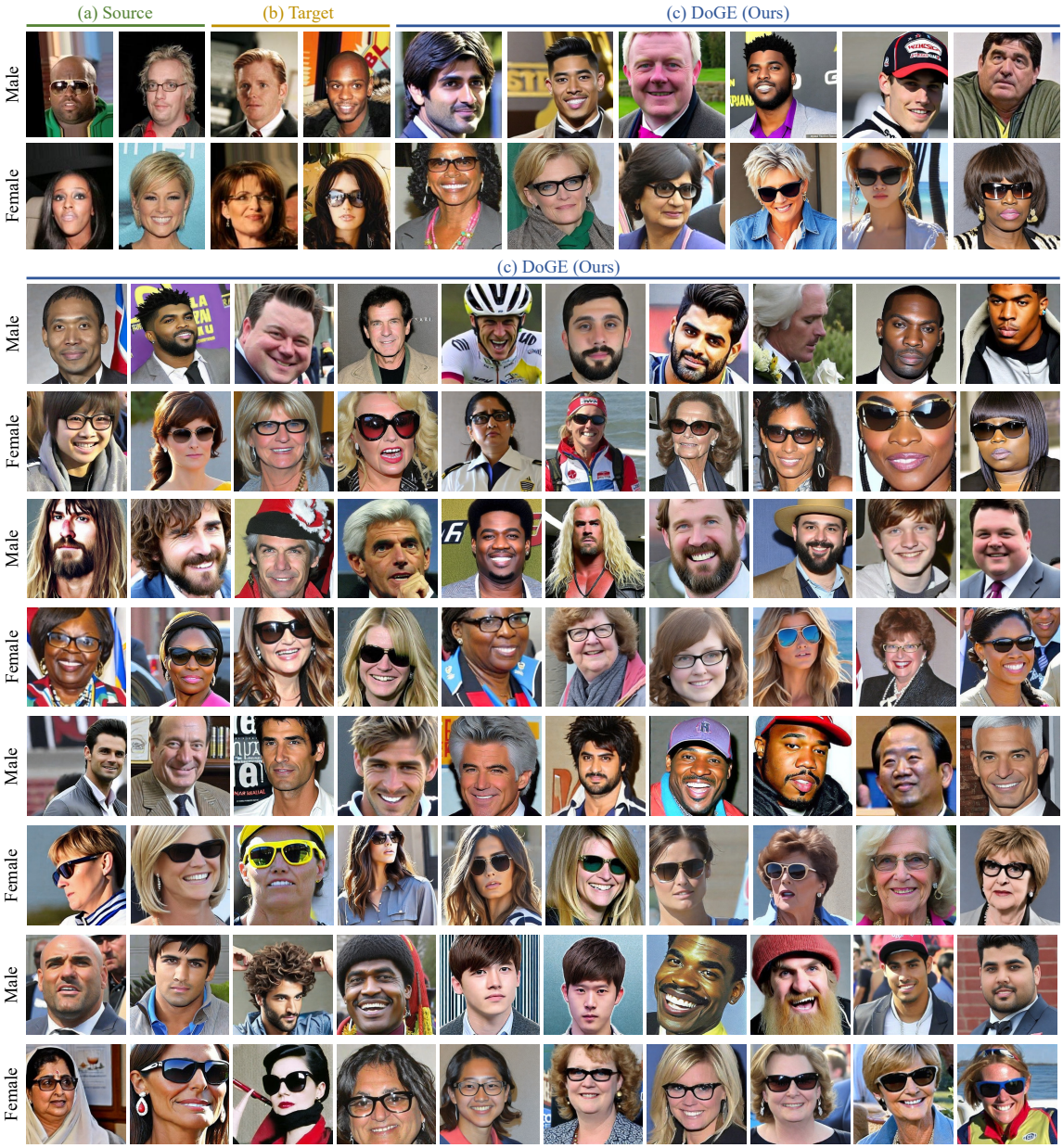

Figure B.4: **More synthetic examples from the CelebA subpopulation shift experiment.** On the top-left, we show the (a) source and (b) target distribution as defined in our experiment setup. The rest of images (c) are synthetic data generated from DoGE. These examples illustrate our capability of capturing and applying semantic distribution gaps.

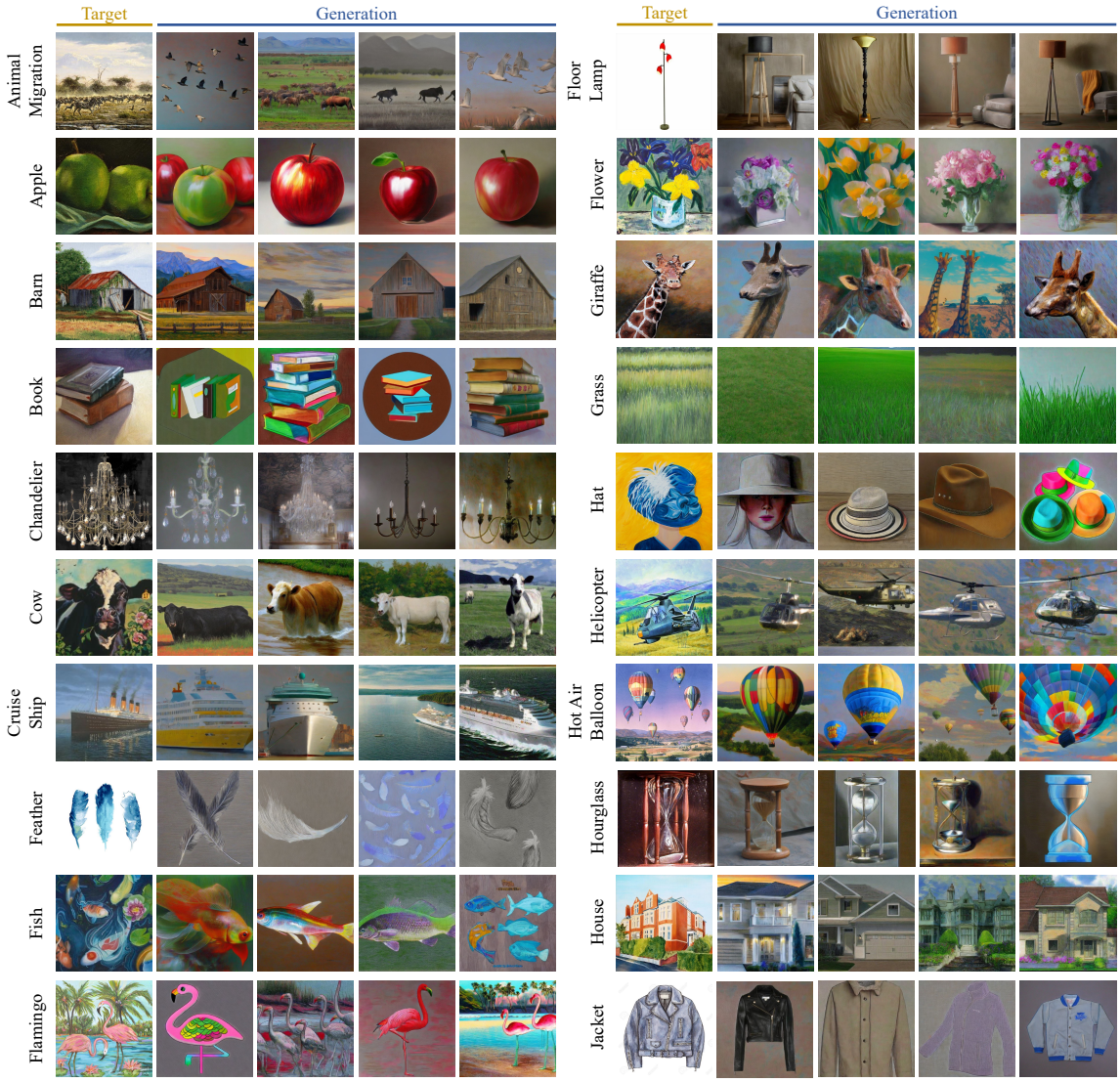

Figure B.5: **More synthetic examples from the DomainNet Real → Painting generation.** We list more synthetic data generated in our Real→Painting UDA experiment in Sec. 3.4.3. We randomly select and show 20 classes from DomainNet. For each class, we present one image from the DomainNet Painting domain as a reference and four of our generations. These examples demonstrate our generation quality and capability to effectively augment real images into the Painting domain.

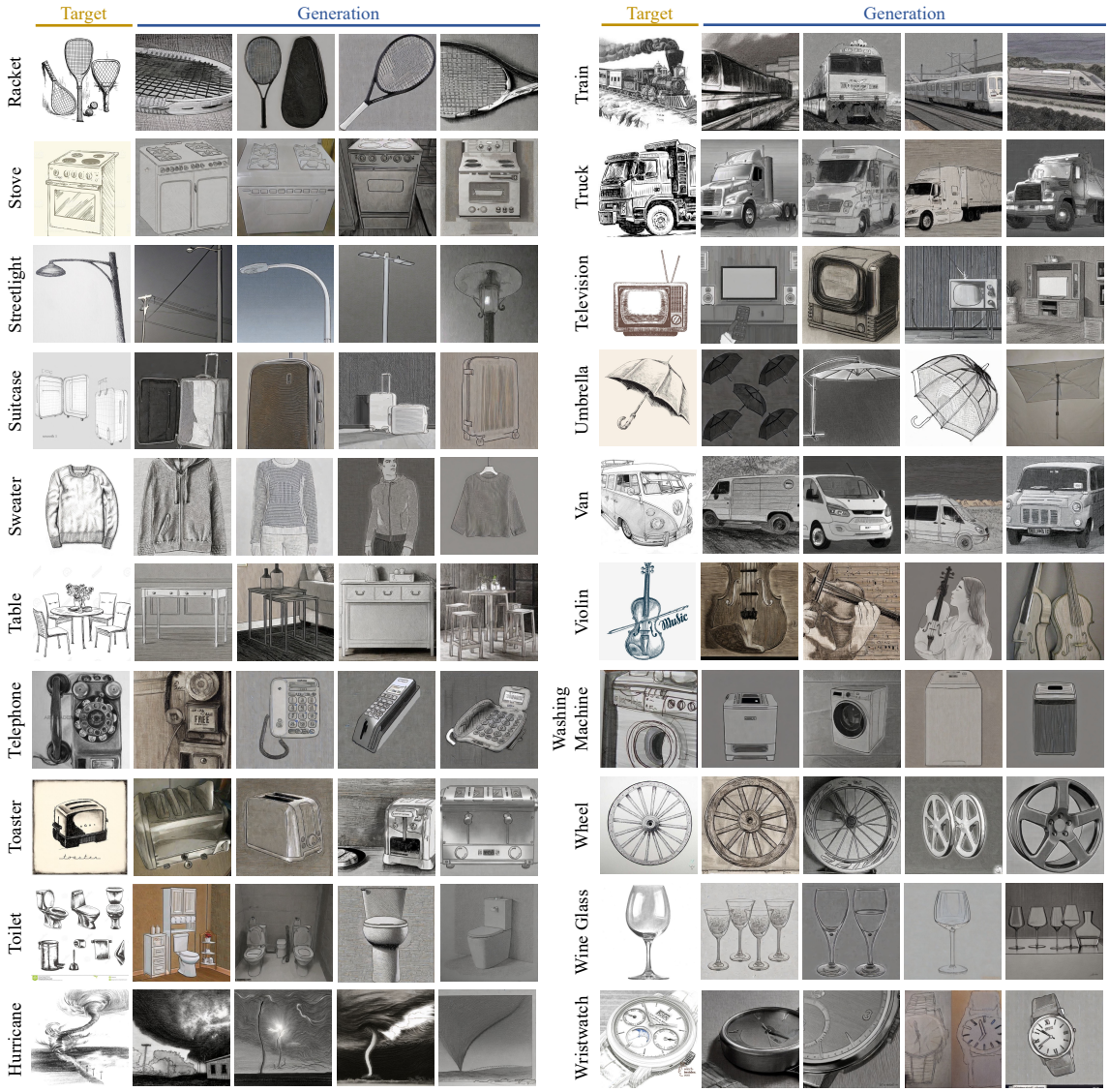

Figure B.6: **More synthetic examples from the DomainNet Real → Sketch generation.** We list more synthetic data generated in our Real→Sketch UDA experiment in Sec. 3.4.3. We randomly select and show 20 classes from DomainNet. For each class, we present one image from the DomainNet Sketch domain as a reference and four of our generations. These examples demonstrate our generation quality and capability to effectively augment real images into the Sketch domain.

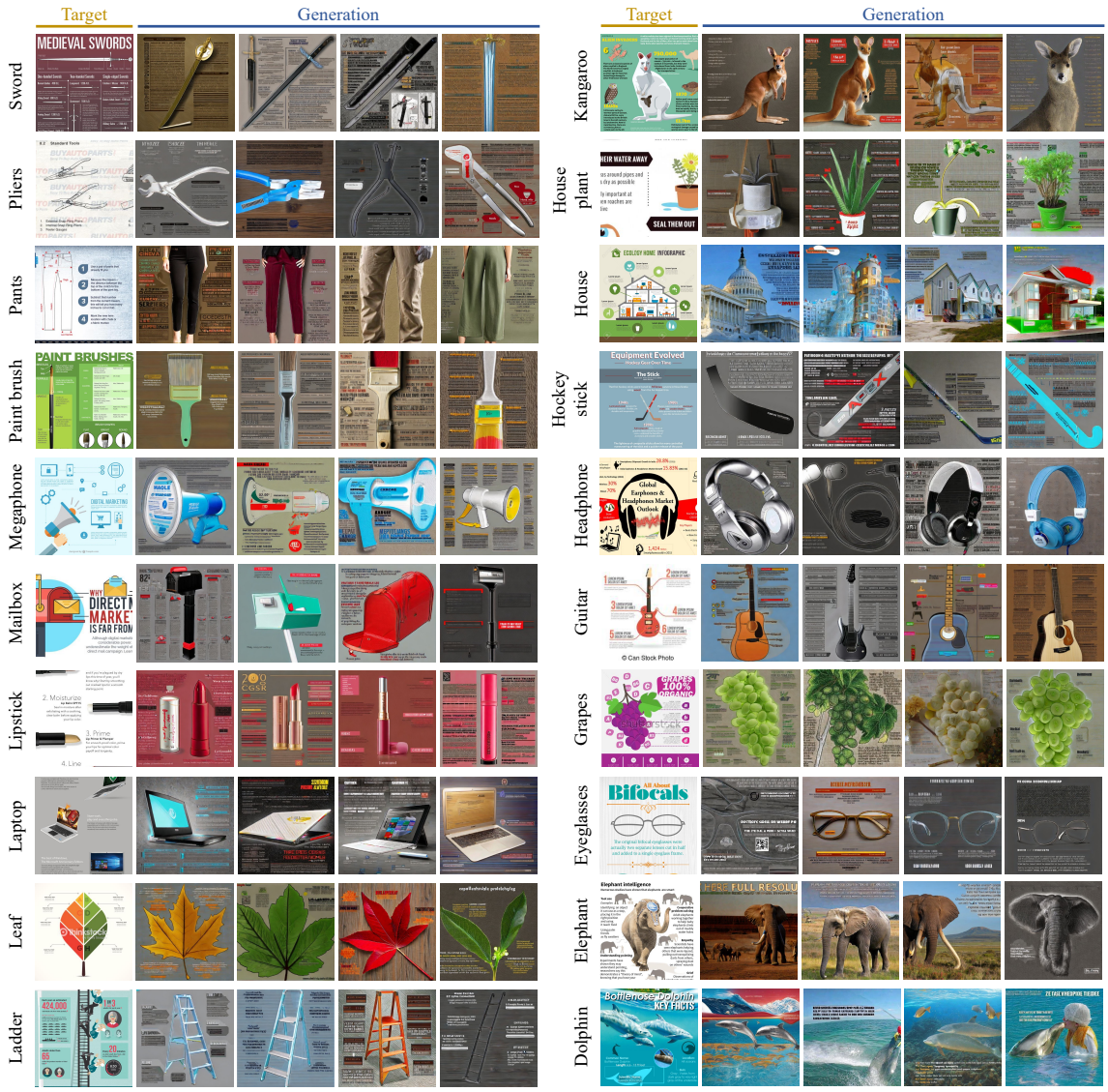

Figure B.7: **More synthetic examples from the DomainNet Real → Infograph generation.** We list more synthetic data generated in our Real→Infograph UDA experiment in Sec. 3.4.3. We randomly select and show 20 classes from DomainNet. For each class, we present one image from the DomainNet Infograph domain as a reference and four of our generations. These examples demonstrate our generation quality and capability to effectively augment real images into the Infograph domain.

Figure B.8: **More synthetic examples from the FMoW domain adaptation experiment.** Following our experiment setup, we augment satellite images from relatively older periods into more recent times. We randomly select and present 10 classes of land use. For each class (row), the leftmost column shows randomly selected references from the target domain; the remaining nine images are our synthetic data.

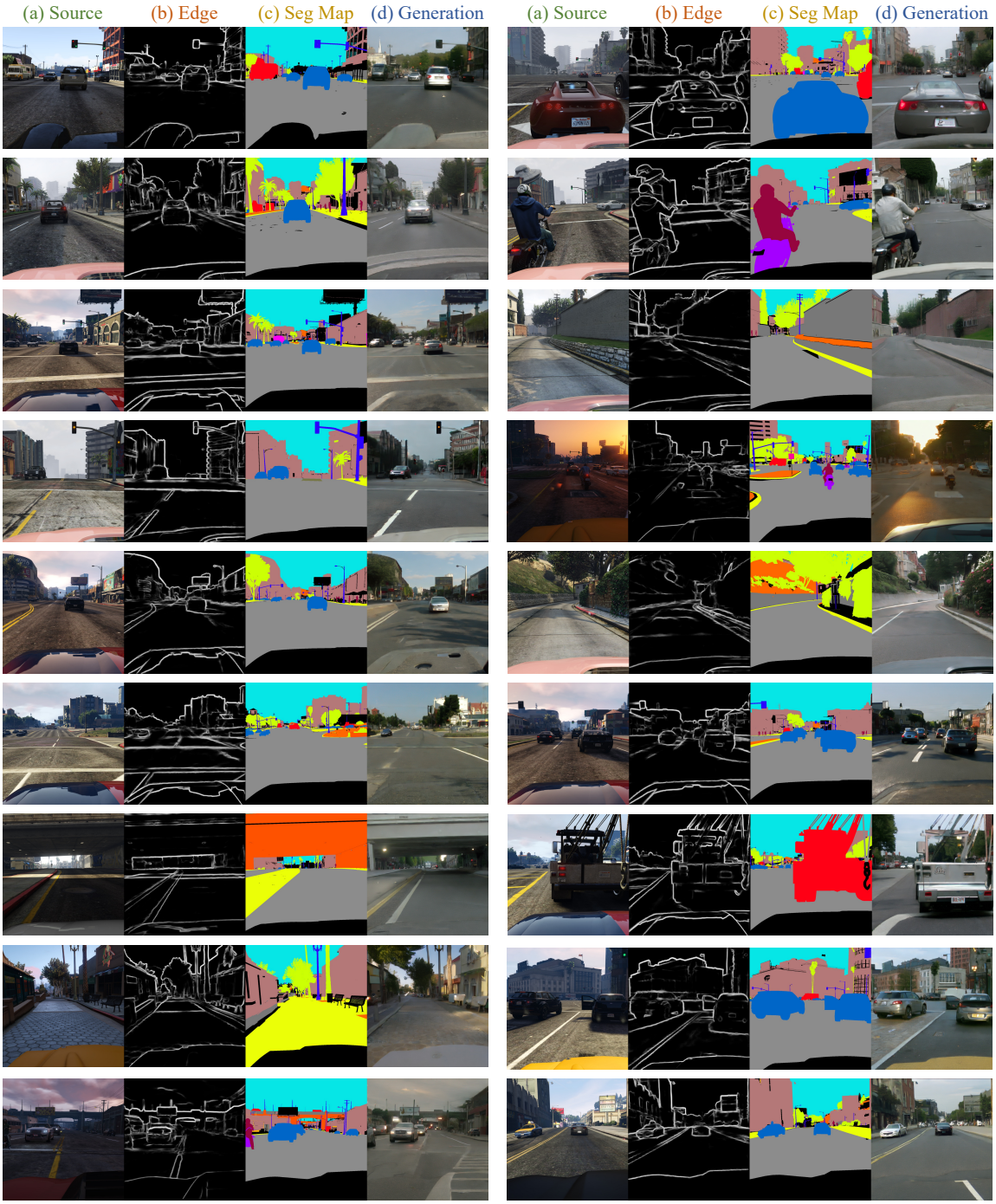

Figure B.9: **More synthetic examples from the GTA5 → CityScapes segmentation experiment.** Given the (a) source data from GTA5, we extract (b) the corresponding edge map. Together with the provided (c) segmentation map, DoGE generated (d) synthetic images that are closer to the CityScapes data distribution. These examples showcase our generation capability for complex scenes.